Humboldt-Universität zu Berlin

Institut für Bibliotheks- und Informationswissenschaft

MASTERARBEIT

Extraktion von RDF-Tripeln aus unstrukturierten Wikipedia-Texten

DBpedia erweitern durch Auswertung kompletter Artikeltexte

Philosophische Fakultät I

Alexander Meyer
alexander.meyer@student.hu-berlin.de

Gutachter: Dipl.-Math. Michael Heinz
Prof. Dr. Stefan Gradmann

Datum der Einreichung: 06.12.2012

Zusammenfassung

Im Projekt DBpedia werden unter anderem Informationen aus Wikipedia-Artikeln in RDF-Tripel umgewandelt. Dabei werden jedoch nicht die Artikeltexte berücksichtigt, sondern vorrangig die sogenannten Infoboxen, die Informationen enthalten, die bereits strukturiert sind. Diese Arbeit stellt WIKI2RDF vor, eine Software zur regelbasierten Extraktion von RDF-Tripeln aus den unstrukturierten Volltexten der Wikipedia. Die Extraktion erfolgt nach Syntax-Parsing mithilfe eines Dependency-Parsers. Die extrahierten Tripel sollen der Erweiterung der DBpedia dienen. Exemplarisch wird WIKI2RDF auf 68820 Artikel aus der Kategorie «Wissenschaftler» der deutschsprachigen Wikipedia angewandt. Es werden 244563 Tripel extrahiert.

Abstract

Extracting RDF statements from unstructured Wikipedia texts: augmenting DBpedia by exploiting full article texts

DBpedia is a project that among other things extracts RDF statements from articles of Wikipedia. However, it does not exploit full article texts but mainly the so-called infoboxes that contain information that is already structured. This thesis introduces WIKI2RDF, a tool for rule-based extraction of RDF triples from full unstructured Wikipedia article texts. Extraction is carried out after syntactical parsing using a dependency parser. Extracted triples are meant to enhance DBpedia. WIKI2RDF is used for triple extraction from 68820 articles about scientists and humanists (category “Wissenschaftler”) in German Wikipedia. 244563 triples are extracted.

Vorbemerkung

Dieser Arbeit liegt eine CD-ROM bei, die folgende Verzeichnisse und Dateien enthält, die nicht im Rahmen dieser Druckfassung abgebildet werden konnten:

`wiki2rdf.pl`	Perl-Skript WIKI2RDF
`kategorie-wissenschaftler.txt`	alphabetische Liste der Artikel der Wikipedia-Kategorie «Wissenschaftler»
`wissenschaftler.nq`	extrahierte vollständige Tripel aus Artikeln der Wikipedia-Kategorie «Wissenschaftler»
`unresolved_anaphora.nq`	extrahierte Tripel mit nicht aufgelösten Anaphern
`rules-wissenschaftler.txt`	Extraktionsregeln für Wissenschaftler
`classifiers-wissenschaftler.txt`	Klassifizierungsregeln für Wissenschaftler
`coreferences.yaml`	Definitionen von Anaphern
`queries/`	SPARQL-Abfragen und Ergebnisse
`Masterarbeit_A_Meyer.pdf`	Druckfassung der Arbeit

An den entsprechenden Textstellen werden die Inhalte der Verzeichnisse und Dateien näher erläutert.

Danksagung

Mein Dank gilt Michael Heinz und Guido Tschacher vom Institut für Bibliotheks- und Informationswissenschaft für die freundliche und unkomplizierte Bereitstellung von Hardware und Speicherplatz zur Programmierung von WIKI2RDF und Anwendung auf große Datenmengen.

Inhaltsverzeichnis

1 Einleitung

1.1 DBpedia und wiki2rdf

DBpedia[1] (Bizer u. a., 2009) ist ein Projekt zur Umwandlung von Informationen aus der Wikipedia in ein Format, das den Prinzipien des Semantic Web bzw. von Linked Data folgt.[2] Dies umfasst im Kern zwei Dinge: Zum einen bietet DBpedia dereferenzierbare URIs für alle Entitäten, die in der Wikipedia beschrieben werden – vulgo einen Linked-Data-tauglichen Identifier für jeden Wikipedia-Artikel. Zum anderen werden im DBpedia-Projekt automatisch nach festgelegten Regeln Aussagen aus der Wikipedia extrahiert und in RDF-Tripel umgewandelt. Dies ermöglicht eine Suche in Wikipedia-Inhalten, die über die traditionelle Volltextsuche weit hinausgeht. Unter anderem können an das DBpedia-Dataset folgende Suchanfragen gestellt werden (nach Jentzsch (2009)):

- alle Bücher von Autoren, die im 19. Jahrhundert in Berlin geboren wurden
- alle Flüsse, die in den Rhein fließen und länger als 50 Kilometer sind
- alle Schauspieler der TV-Serie «Lost», die in England geboren wurden
- alle Tennisspieler aus Moskau
- alle Fußballspieler mit der Rückennummer 11, die für einen Verein spielen, der ein Stadion mit über 40000 Plätzen hat, und die in einem Land geboren wurden, das mehr als 10 Millionen Einwohner hat

DBpedia bietet unter anderem einen SPARQL-Endpoint zur Suche an. Die zum Zeitpunkt dieser Arbeit aktuelle Version von DBpedia ist 3.8 vom August 2012 und enthält der englischsprachigen Wikipedia entnommene Aussagen über 3.77 Millionen

[1] `http://dbpedia.org/`

[2] Was das Semantic Web ist und wie es in technischer Hinsicht funktioniert, soll an dieser Stelle nicht erläutert werden. Es sei auf die inzwischen zahlreichen Einführungstexte verwiesen, z. B. Hitzler u. a. (2008). Ebenso wird auf eine Einführung in das Wesen und die Geschichte der Wikipedia verzichtet, vgl. hierzu bspw. die Wikipedia selbst: `http://de.wikipedia.org/wiki/Wikipedia`. Auch werden an dieser Stelle nicht die Bedeutung der Begriffe «Information» und «Wissen» diskutiert sowie die Frage, ob z. B. in RDF-Tripeln Information oder Wissen oder beides steckt. Es wird folgende Terminologie festgelegt: Texte sowie RDF-Tripel enthalten *Informationen* bzw. *Aussagen*. Es genügt für die Zwecke dieser Arbeit, die Begriffe «Information» und «Aussage» als synonym zu betrachten.

Dinge[3], wovon 2.35 Millionen in der zugehörigen DBpedia-Ontologie klassifiziert sind, darunter 764000 Personen, 573000 Orte, 333000 künstlerische Arbeiten, 192000 Körperschaften, 202000 Tierarten und 5500 Krankheiten (Bizer, 2012). Seit DBpedia 3.7 werden neben der englischsprachigen Wikipedia auch andere Sprachvarianten berücksichtigt und zur Tripelextraktion verwendet, unter anderem existiert seitdem der Ableger DBpedia Deutsch[4], dem die deutschsprachige Wikipedia zugrunde liegt.

Die DBpedia-Ontologie[5] umfasst in Version 3.8 359 Klassen, 1775 Eigenschaften und – wie erwähnt – 2.35 Millionen Instanzen aus der englischsprachigen Wikipedia. Hinzu treten ggfs. weitere Klassen, Eigenschaften und Instanzen aus den anderen Sprachvarianten.

Der Prozess der Tripelextraktion wird detailliert in Auer und Lehmann (2007) beschrieben. Dabei fällt auf, dass mitnichten die gesamte Wikipedia bzw. komplette Wikipedia-Artikel zur Extraktion herangezogen werden, obgleich dies in einer zusammenfassenden Darstellung zur DBpedia angedeutet wird.[6] Die Extraktion beschränkt sich hauptsächlich auf die sogenannten Infoboxen. Dabei handelt es sich um tabellarische Darstellungen, die in der Regel am Beginn eines Artikels erscheinen und Aussagen über das Thema in Form von Attribut-Wert-Paaren enthalten. Infoboxen sind je nach Typ der Entität, die im Artikel beschrieben wird (Ort, Person, chemisches Element ...), unterschiedlich ausgeprägt. Abb. 1 zeigt die Infobox zur Stadt Hannover aus der deutschsprachigen Wikipedia in Form von Wikitext[7] sowie in ihrer Darstellung in der Weboberfläche.

Es fällt auf, dass die Informationen bereits in einer stark strukturierten Form sowie in teilkontrolliertem Vokabular vorliegen, sodass sie mit geringem Aufwand in RDF-Tripel umgewandelt werden können: Aus der Aussage `Bundesland = Niedersachsen` im Artikel «Hannover» kann schnell das Tripel

```
ex:Hannover exprop:liegt_in ex:Niedersachsen .
```

(hier mit beispielhaften Namespaces) werden. Die Informationen, die in DBpedia extrahiert werden, sind also solche, die bereits in der Wikipedia als Quasi-Tripel existieren und lediglich eine neue Syntax bekommen.[8]

[3] In dieser Arbeit steht der englische Dezimalpunkt anstelle des deutschen Kommas zur Trennung von Dezimalstellen in Zahlen.

[4] `http://de.dbpedia.org/`

[5] `http://wiki.dbpedia.org/Ontology`

[6] «We develop an information extraction framework that *converts Wikipedia content* into a rich multi-domain knowledge base.» (Bizer u. a., 2009, Preprint S. 2, Hervorhebung v. Autor)

[7] Wikitext ist die Markup-Sprache, in der Texte in Wikis geschrieben werden.

[8] Aufwendiger ist naturgemäß das Erstellen einer konsistenten Ontologie anhand der Eigenschaften und Werte in den Infoboxen, denn diese wurden nicht zu dem Zweck der Ontologiebildung geschaffen. Dies, also die Formulierung der Klassen und Eigenschaften der DBpedia-Ontologie, ist daher eine Kernaufgabe des DBpedia-Projektes.

Wappen	Deutschlandkarte

Basisdaten	
Bundesland:	Niedersachsen
Landkreis:	Region Hannover
Höhe:	55 m ü. NN
Fläche:	204,1 km²
Einwohner:	525.875 (31. Dez. 2011)[1]
Bevölkerungsdichte:	2576 Einwohner je km²
Postleitzahlen:	30159–30659
Vorwahl:	0511
Kfz-Kennzeichen:	H
Gemeindeschlüssel:	03 2 41 001
Stadtgliederung:	13 Stadtbezirke, 51 Stadtteile
Adresse der Stadtverwaltung:	Trammplatz 2 30159 Hannover
Webpräsenz:	www.hannover.de
Oberbürgermeister:	Stephan Weil (SPD)

```
{{Infobox Gemeinde in Deutschland
|Art               = Stadt
|Name              = Hannover
|Wappen            = Coat of arms of Hannover.svg
|Breitengrad       = 52/22/28
|Längengrad        = 9/44/19
|Bundesland        = Niedersachsen
|Landkreis         = Region Hannover
|Höhe              = 55
|Fläche            = 204.14
|PLZ               = 30159-30659
|Vorwahl           = 0511
|Kfz               = H
|Gemeindeschlüssel = 03241001
|LOCODE            = DE HAJ
|Gliederung        = 13 [[Stadtbezirk]]e, 51 [[Stadtteil]]e
|Adresse           = Trammplatz 2<br />30159 Hannover
|Website           = [http://www.hannover.de/ www.hannover.de]
|Bürgermeister     = [[Stephan Weil]]
|Bürgermeistertitel= Oberbürgermeister
|Partei            = SPD
}}
```

Abbildung 1: Infobox zu Hannover (`http://de.wikipedia.org/w/index.php?title=Hannover&oldid=110205704`) als Wikitext und in der Anzeige (ohne Lageplan; die Einwohnerzahl enstammt einer externen Quelle und erscheint nicht im Quelltext des Artikels)

Zur Gewinnung weiterer Tripel liegt es daher nahe, die Extraktion nicht auf die vorstrukturierten Teile der Wikipedia-Artikel zu beschränken, sondern andere hinzuziehen, allen voran die Artikeltexte selbst. Es ist davon auszugehen, dass diese in einem hohen Maße weitere Aussagen enthalten, die als Tripel formuliert werden können und nicht durch die Infoboxen abgedeckt werden.

Man stelle sich bspw. die Biographie eines Wissenschaftlers vor, die mit hoher Wahrscheinlichkeit Angaben dazu enthält, wann er wo und welches Fach studiert hat, wo er arbeitet und welche Auszeichnungen er erhalten hat. Diese Angaben sind oftmals nicht über Infoboxen oder andere vorstrukturierte Mittel (z. B. die Kategorien, denen ein Artikel zugeordnet ist) zu gewinnen, sondern stecken in ausformulierten Sätzen im Volltext von Artikeln.

Die vorliegende Arbeit stellt die Software WIKI2RDF vor, ein Perl-Skript, das regelbasiert Satzmuster in Wiki-Texten in RDF-Aussagen (Tripel bzw. Quadrupel) umwandelt. Regeln für das Mapping zwischen einer natürlichsprachlichen Formulierung und deren RDF-Äquivalent können mithilfe einer einfachen Syntax formuliert werden, auch ohne tief gehende Kenntnisse von WIKI2RDF. Analog zu DBpedia bietet es sich an, für verschiedene Typen von Entitäten (Orte, Personen, chemische Elemente ...) unterschiedliche Sets von Mapping-Regeln zu entwerfen. Beispielhaft werden in dieser Arbeit Regeln für Artikel der Kategorie «Wissenschaftler» entworfen und diese auf die Artikel über Wissenschaftler der deutschsprachigen Wikipedia angewandt. Es gelang, mithilfe von 150 Regeln 244563 vollständige Tripel aus jenen

Artikeln zu extrahieren. Die Ergebnisse werden hier diskutiert.

Von WIKI2RDF extrahierte Tripel folgen in ihrer Grundstruktur den Datasets von DBpedia. Die Arbeitsergebnisse sind also als Erweiterung bzw. Anreicherung der bestehenden DBpedia zu verstehen.

WIKI2RDF arbeitet auf Basis morphologisch-syntaktischer Satzanalysen, die derzeit vom Syntax-Parser PARZU durchgeführt werden, der eine Dependenz-Grammatik für das Deutsche realisiert.[9] Daher können derzeit nur Texte in deutscher Sprache bearbeitet werden. Prinzipiell ist es jedoch möglich, WIKI2RDF andere Sprachen beizubringen, sofern hierfür Tools vorliegen, die linguistische Analysen in ähnlicher Form wie PARZU erbringen.

1.2 Verwandte Arbeiten

Als verwandt mit der vorliegenden Arbeit können in erster Linie solche gelten, die sich ebenfalls mit der (linguistischen) Analyse von Wikipedia-Volltexten bzw. der Extraktion von Aussagen hieraus befassen. Nach Kenntnisstand des Autors sind dies, gemessen an der gesamten Wikipedia-Forschung innerhalb der Informationswissenschaft, Computerlinguistik und Informatik, sehr wenige. Nakayama u. a. (2008, Abschn. 3.4) kündigen eine «Wikipedia Ontology» an, die, soweit erkennbar, mit ähnlichen Methoden erarbeitet werden soll. Nakayama (2008) beschreibt dieses Konzept näher, jedoch ist die Ontologie nach wie vor nicht realisiert[10]. Herbelot und Copestake (2006) arbeiten ebenfalls an der Extraktion von Aussagen und mithilfe eines Dependency-Parsers, beschränken sich jedoch auf Hyperonymie-Hyponymie-Relationen. Relationen dieser Natur sind allerdings in einem hohen Maße schon in den strukturierten Teilen der Wikipedia gegeben – etwa beschreiben die Kategorien, denen ein Artikel zugeordnet ist, oftmals taxonomische Relationen zwischen Kategorie und Artikel – und müssen nicht notwendigerweise aus den Volltexten gewonnen werden. Nguyen u. a. (2007) beschreiben ein System ähnlich dem in dieser Arbeit vorgestellten, beschränken sich jedoch auf wenige Typen von Relationen und zudem ebenfalls auf solche, die leicht auch anhand der strukturierten Artikelteile hätten gewonnen werden können: Abb 4 auf S. 1419 von Nguyen u. a. (2007) zeigt einige der extrahierten Relationen – es sind sämtlich solche, für die die Volltexte nicht hätten konsultiert werden müssen.

Keine der genannten Arbeiten geht über beispielhafte Extraktionen hinaus und liefert einen Rahmen, um beliebige Extraktionen z. B. anhand von Mapping-Regeln

[9] PARZU benutzt TREETAGGER zur morphologischen Analyse, Grundformenreduktion und Wortartenbestimmung (Part-of-Speech-Tagging).

[10] Vgl. `http://sigwp.org/en/index.php/Wikipedia_API`, wo die «Wikpedia Ontology» als «coming soon» bezeichnet wird (letzter Zugriff 3. Dezember 2012).

vornehmen zu können. Dies leistet die vorliegende Arbeit. Zudem beschränken sich alle zitierten Arbeiten auf die englischsprachige Wikipedia.

Daneben haben Atserias u. a. (2008) Wikipedia-Artikel syntaktisch und semantisch analysiert[11], machen jedoch keine Aussagen über realisierte Anwendungen ihrer Ergebnisse.

Die vorliegende Arbeit kann zudem als Gegenstück zur Software «Semantic MediaWiki»[12] (Krötzsch u. a., 2007) betrachtet werden: Semantic MediaWiki erlaubt das manuelle Einfügen von RDF-Tripeln in Wikitexte, diese Arbeit versucht, sie automatisch zu generieren.

In einem größeren Kontext ist diese Arbeit naturgemäß mit allen verwandt, die die Extraktion strukturierter Aussagen aus un- oder semi-strukturierten natürlichsprachlichen Volltexten zum Ziel haben, ob allgemein als Informationsextraktion (*information extraction*) oder, mit dem Ziel der Ontologiebildung, als *ontology learning from text*. Auf einzelne Arbeiten soll hier nicht verwiesen werden, vgl. als Einführung in die Informationsextraktion etwa Jurafsky und Martin (2009, S. 759 – 798), zu *ontology learning from text* Buitelaar u. a. (2005) mit vielen Literaturhinweisen.

[11] `http://barcelona.research.yahoo.net/dokuwiki/doku.php?id=semantically_annotated_snapshot_of_wikipedia`

[12] `http://semantic-mediawiki.org/`

2 Vorüberlegungen

2.1 Schwierigkeiten der Informationsextraktion aus Texten

Es liegt auf der Hand, dass die Extraktion von Informationen bzw. in diesem Fall konkret die Extraktion von RDF-Tripeln aus Texten[13] allgemein mit verschiedenen Schwierigkeiten zu kämpfen hat, die in der Natur natürlichsprachlicher Texte liegen. Diese können hier nicht detailliert diskutiert, sondern lediglich knapp angerissen werden. Zur Vertiefung sei erneut auf Jurafsky und Martin (2009, S. 759 – 798) verwiesen. Abschnitt 2.2 dieser Arbeit beschreibt, ob und in welcher Form diese in Wikipedia auftreten. Es handelt sich in den germanischen und vielen indoeuropäischen Sprachen mindestens um folgende Phänomene:[14]

1. Die gleiche Entität kann unterschiedlich benannt werden.

2. Auf bereits benannte Entitäten oder ganze Aussagen kann mithilfe von Anaphern Bezug genommen werden («Er hat sie ihm gegeben.» – «Das glaube ich nicht.» – «Doch, es war so.»)

3. Die (grammatikalischen) Rollen von Entitäten in Aussagen sind nicht markiert durch Pfeile oder Ähnliches, sondern ergeben sich aus der morphologisch-syntaktischen Struktur eines Satzes: Flexionsendungen, Wortstellung, Präpositionen usw. deuten auf diese hin, sind jedoch unverständlich ohne vorheriges Wissen über deren Bedeutung.

4. Aus Punkt 3 ergibt sich, dass ein Lemma z. B. durch Flexionsendungen unterschiedliche Gestalt annehmen kann. («das Haus» – «des Hauses» – «die Häuser» – «den Häusern»)

5. Aussagen können mit unterschiedlichem Vokabular gemacht werden. («Er war von 1980 bis 1985 als Student an der Universität Köln eingeschrieben.» – «Er studierte von 1980 bis 1985 an der Universität zu Köln.» – «Er begann sein Studium an der Kölner Universität 1980 und schloss es 1985 ab.»)

6. Es existieren Haupt- und Nebensätze sowie andere Satz- und Teilsatzstrukturen mit unterschiedlicher Syntax. Gesuchte Aussagen können sich in verschiedensten syntaktischen Strukturen verstecken.

[13] Die «Extraktion von RDF-Tripeln» bedeutet natürlich, dass zunächst eine Information aus einem Text extrahiert werden muss, die dann als RDF-Tripel ausgedrückt wird, und nicht, dass direkt ein Tripel extrahiert wird. Vereinfachend wird im Verlauf dieser Arbeit jedoch von «Extraktion von RDF-Tripeln» die Rede sein, um den sprachlichen Ausdruck nicht unnötig zu verkomplizieren.

[14] Andere Sprachfamilien sollen hier nicht behandelt werden.

7. Trotz standardisierter Rechtschreibung und Grammatik erscheinen in der Realität orthographische und grammatikalische Fehler.

Punkt 1 wird begegnet durch *named entity recognition*, also die Identifikation von Entitäten und inbesondere Eigennamen in einem Text. Eng damit verknüpft sind die *anaphora resolution* (Anaphernresolution) bzw. allgemeiner die *coreference resolution*, die Punkt 2 betreffen. Anaphernresolution bedeutet die Rückführung eines anaphorischen Ausdrucks auf seinen Bezugsausdruck (Antezedens).

Den Punkten 3, 4 und in Teilen 6 wird durch verschiedene Hilfsmittel begegnet, die auf verschiedenen Ebenen arbeiten. Auf jeden Fall benötigt wird Grundformenreduktion, also das Zurückführen der verschiedenen Wortformen auf ihre Grundform (das zugehörige Lemma) oder eine Stammform. Verfahren hierzu existieren z. B. im Kontext des Information Retrieval in großer Zahl (dort bezeichnet als Teil der «Automatischen Indexierung») und sollen hier nicht näher beschrieben werden. Es sei zur Einführung auf Gödert u. a. (2012, S. 245 – 326), Nohr (2005) und nach wie vor auf Kuhlen (1977) verwiesen.[15] Damit zusammen hängt die Bestimmung der Wortart (*part of speech*) jedes Wortes (Jurafsky und Martin, 2009, S. 157 – 206). Dies leisten sogenannte Part-of-Speech-Tagger (PoS-Tagger). In der vorliegenden Arbeit angewandt wurde TreeTagger, der mit verschiedenen Sprachen umgehen kann und sowohl die Wortart als auch das Lemma ausgibt.

Auf dieser Basis können nun die syntaktische Struktur eines Satzes und damit die grammatikalischen Rollen, die nicht notwendigerweise mit den logischen bzw. realen Rollen übereinstimmen müssen, näher analysiert werden. Das sogenannte Syntax-Parsing kann verschieden tief, also verschieden genau, durchgeführt werden. Zudem existieren verschiedene Grammatiken, d. h. syntaktische Modelle, auf die ein Satz zurückgeführt wird, etwa Kontextfreie Grammatiken (*context-free grammar*) und Dependenz-Grammatiken (*dependency grammar*). Im Gegensatz zu Kontextfreien Grammatiken abstrahieren Dependenz-Grammatiken von der Wortstellung im Satz und liefern lediglich syntaktische oder semantische Relationen zwischen Wörtern. Dabei wird oftmals das Verb als Kopf des Satzes begriffen, von dem die anderen Teile abhängen bzw. diese eine bestimmte Beziehung zum Verb haben. Daher eignen sich Dependenz-Grammatiken für Sprachen mit freier Wortstellung wie das Deutsche. Die in dieser Arbeit verwendete Software ParZu ist dementsprechend ein Dependency-Parser.

Punkt 5 aus obiger Liste wird in dieser Arbeit mithilfe manuell formulierter Regeln begegnet, wie in den folgenden Kapiteln dargestellt. Punkt 7 wird hier nicht

[15] Vor der Grundformenreduktion muss natürlich noch identifiziert werden, was ein Wort überhaupt ist, wo es anfängt und endet. Die genannte Literatur bietet auch hierzu Hinweise, unter den Überschriften Tokenisierung und Normalisierung.

näher behandelt, da er in dieser Arbeit durch die Natur der analysierten Texte (Lexikonartikel) keine nennenswerte Rolle spielt.

2.2 Besonderheiten der Wikipedia

Die im vorigen Abschnitt dargestellten Punkte erfahren, bezogen auf Wikipedia-Texte, einige Vereinfachungen.

Zunächst ist davon auszugehen, dass Lexikonartikel generell in einem Stil formuliert sind, der die Extraktion von Tripeln vereinfacht. Lexikonartikel sind um eine klare Ausdrucksform, übersichtliche und nicht zu lange Sätze sowie allgemeine Verständlichkeit bemüht. Größtmögliche sprachliche Vielfalt ist nicht ihr Ziel.

Dies bedeutet neben einfacherem Syntax-Parsing, dass die Zahl und Natur von Anaphern beschränkt ist: In computerlinguistischer Einführungsliteratur zur Anaphernresolution werden oft komplexe Beispiele etwa aus Zeitungsartikeln gegeben, die darum bemüht sind, fokussierte Entitäten auf möglichst viele Weisen zu referenzieren, um nicht durch mangelnde sprachliche Vielfalt uninteressant zu wirken. So kann ein Zeitungsartikel über den fiktiven Politiker Max Mustermann (SPD-Mitglied, 57 Jahre alt, ehemaliger Bürgermeister von Musterstadt und jetzt Innenminister) diesen als «Max Mustermann», «Mustermann», «er», «der Politiker», «der Innenminister», «der Minister», «der ehemalige Bürgermeister von Musterstadt», «der frühere musterstädtische Bürgermeister», «der SPD-Politiker», «das SPD-Mitglied», «der 57-Jährige» usw. bezeichnen. Es erfordert nicht nur Wissen über Wortformen, sondern im Hintergrund auch Weltwissen, diese Anaphern sämtlich nach «Max Mustermann» aufzulösen. Ein solcher Schreibstil würde in einem biographischen Lexikonartikel jedoch als unangemessen empfunden. Es ist davon auszugehen, dass in Wikipedia-Artikeln Anaphern zum einen nicht viele Formen haben, sondern größtenteils Pronomina sind, und sie sich zum anderen meist auf das Lemma des Artikels beziehen. Dies vereinfacht die Anaphernresolution erheblich.

Daneben müssen Rechtschreib- und Grammatikfehler nicht gesondert berücksichtigt werden, da sie nicht generell Teil von Lexikonartikeln sind.

Ein weiterer Vorteil ergibt sich aus der Natur der Wikipedia: Wikitexte sind Hypertexte, sie enthalten Links zu anderen Teilen des Wikis, im Fall von Wikipedia also zu anderen Artikeln. Abb. 2 zeigt ein Beispiel aus dem Quelltext des Artikels «Köln». Links sind durch Paare eckiger Klammern (`[[`, `]]`) markiert und erscheinen in der Webansicht als HTML-Links. Damit sind *named entities* bereits gegeben und müssen nicht von Grund auf identifiziert werden. Der gezeigte Ausschnitt enthält unter anderem die Entitäten «Monte Troodelöh», «Königsforst», «Worringer Bruch», «Kölner Bucht», «Bergisches Land», «Eifel», «Rhein» und «Rheinisches Schieferge-

```
Der [[Topografie (Kartografie)|topographische]] [[Bezugspunkt]] der Stadt, die
Spitze des nördlichen Domturms, liegt 50° 56’ 33’’ nördlicher Breite und 6° 57’
32’’ östlicher Länge. Der höchste Punkt liegt 118,04 Meter (der [[Monte
Troodelöh]] im [[Königsforst]]), der niedrigste 37,5 Meter (im [[Worringer
Bruch]]) über dem Meeresspiegel.

Köln liegt in der [[Kölner Bucht]], einer trichterförmigen, durch den Rhein
geprägten Flusstallandschaft zwischen den stufenartig ansteigenden Hängen des
[[Bergisches Land|Bergischen Landes]] und der [[Eifel]] unmittelbar nach
Austritt des [[Rhein]]s aus dem [[Rheinisches Schiefergebirge|Rheinischen
Schiefergebirge]]. Diese geschützte, günstige Lage bewirkt für Köln ein mildes
[[Klima]], das sich durch mehrere Besonderheiten auszeichnet: (...)
```

Abbildung 2: Wikilinks im Artikel über Köln (`http://de.wikipedia.org/w/index.php?title=Köln&oldid=110189858`)

birge». In der Regel werden diese jedoch nur bei ihrem ersten Auftreten im Artikel markiert. Zur Identifizierung aller Entitäten genügt es daher also nicht, nur die Stellen mit `[[` zu betrachten, sondern es müssen alle Links zunächst extrahiert werden. Danach muss nach ihren Wörten im kompletten Artikel gesucht werden, z. B. also nach «Rhein», um alle Erwähnungen des Rheins im Köln-Artikel zu finden. Die referenzierten Entitäten sollten jedoch alle mindestens einmal durch einen Link markiert sein, was die *named entity recognition* erheblich vereinfacht.

Bezogen auf die Extraktion von Tripeln kann hier schon einen Schritt weiter gegangen und festgelegt werden, dass sich – in der Regel – durch einen solchen Link bereits Subjekt und Objekt des Tripels ergeben. Subjekt ist im Beispiel immer «Köln», Objekt etwa «Rhein», «Königsforst» oder «Eifel», und nun gilt es lediglich, die Prädikate herauszufinden.

3 Konzept von wiki2rdf

Im Folgenden soll erläutert werden, welche grundlegenden Merkmale WIKI2RDF hat und, im Groben, wie diese umgesetzt werden. Der nächste Abschnitt 4 beschreibt das Skript detaillierter.

Ausgehend von den Vorüberlegungen aus dem vorigen Abschnitt sowie in Anlehnung an DBpedia können folgende Merkmale festgehalten werden:

- WIKI2RDF soll Tripelextraktionen anhand von vorher festgelegten Wort- bzw. Phrasenmustern aus den Volltexten der Artikel vornehmen können. Dazu soll eine einfache Regelsprache entworfen werden, in der die Muster festgehalten werden. Die Formulierung der Regeln erfolgt manuell. Die Regelsprache ermöglicht es, auch ohne tiefgehende Kenntnisse von WIKI2RDF oder gar die Bearbeitung des Quelltextes, Regeln für das Mapping von Wortmuster zu Tripel festzuhalten, und ist also unabdingbar für eine breite Anwendung des Tools.[16]

- Wegen der im vorigen Abschnitt beschriebenen mannigfaltigen Flexions- und Wortstellungsmöglichkeiten vieler natürlicher Sprachen – inklusive der Deutschen – wäre es sehr aufwendig, Regeln zu formulieren, die sich auf die reinen Wörter (Terme, Token) im Volltext beziehen. Es ist daher sehr sinnvoll, wenn eine gewisse Vereinheitlichung stattfindet. Deswegen sollen im Vorfeld Lemmatisierung, PoS-Tagging sowie syntaktisches Parsing anhand einer Dependenz-Grammatik erfolgen, auf deren Basis daraufhin die Regeln formuliert werden. Eine beispielhafte Regel könnte also lauten: Wenn das Verb «studieren» lautet, in der Subjektposition ein Personenname steht und in der Objektposition die Präposition «in», gefolgt von einem Ortsnamen, dann gib das Tripel «*Person* `has_studied_in` *Ort*» (hier wie in allen folgenden Beispielen ohne Namespaces) aus.[17]

- Subjekte und Objekte der Tripel sind die durch das DBpedia-Projekt erstellten URI-Repräsentationen der Wikipedia-Artikel. Dies ermöglicht die direkte Einbindung der hier extrahierten Tripel in das DBpedia-Dataset. Prädikate können der DBpedia-Ontologie entstammen, müssen dies aber nicht.

- Bei vielen der extrahierbaren Aussagen (z. B. «X studiert in Y») ist es sinnvoll, zudem eine Zeitangabe festzuhalten. Diese ist oftmals ebenfalls im Wikipedia-

[16] Analog bietet DBpedia das Formulieren von Mappings innerhalb eines Wikis (`http://mappings.dbpedia.org/`), in dem jeder mitmachen kann.

[17] Zu unterscheiden sind hier die Subjekte und Objekte der Sätze von denen der Tripel. Diese müssen nicht notwendigerweise übereinstimmen.

Artikel zu finden. WIKI2RDF soll daher die Möglichkeit bieten, auch Zeitangaben zu extrahieren. Für den zeitlichen Bezug sollen die Tripel zu Quadrupeln erweitert werden, wobei die Zeitangabe in der vierten Position steht.[18]

- Neben den Volltext-Absätzen in den Wikipedia-Artikeln ist es auch sinnvoll, Listen heranzuziehen: Im Artikel zu einem Wissenschaftler kann aus einer Liste mit der Überschrift «Auszeichnungen» geschlossen werden, dass in ihr wissenschaftliche Auszeichnungen und Preise genannt werden, die die Person erhalten hat. Dies ermöglicht Tripel wie «*Person* `has_won_award` *Preis*». Daher soll die Regelsprache nicht nur ganze Sätze, sondern auch Listenumgebungen umfassen.

- Ebenso wie bei DBpedia verschiedene Mapping-Regeln für verschiedene Typen von Infoboxen existieren, ist es auch hier sinnvoll, verschiedene Sets von Regeln für verschiedene Typen von Entitäten (Wissenschaftler, Musiker, Orte, chemische Elemente ...) zu erschaffen.

- Bei der manuellen Formulierung von Regeln ist ja nicht im Vorfeld klar, welche Formulierungen in den Artikeln überhaupt erscheinen. Daher benötigt WIKI2RDF einen Vorschlagsmodus, in dem ausgegeben wird, welche Verben, Subjekte und Objekte in welchen Kombinationen in den Artikeln überhaupt erscheinen. Anhand dessen können dann die Regeln formuliert werden. Es ist sinnvoll, wenn der Vorschlagsmodus auf eine Stichprobe aus allen Artikeln angewendet wird, aus denen später die Tripel extrahiert werden.

Naturgemäß benötigt WIKI2RDF zudem die Wikipedia-Artikel selbst. Diese können als Dump im XML-Format heruntergeladen werden (siehe S. 39), wobei XML lediglich als Wrapper fungiert: die Artikel selbst liegen nach wie vor im Wikitext-Format vor, demselben Format, in dem die Artikel auch editiert wurden.

In der vorliegenden Arbeit wurde WIKI2RDF nur für die deutsche Sprache und damit für die deutschsprachige Wikipedia umgesetzt. Es ist die Frage erlaubt, warum WIKI2RDF nicht für das Englische entworfen wurde.

Dies hat zum einen den rein praktischen Grund, dass die englischsprachige Wikipedia weitaus größer als die deutsche ist.[19]. Da allein Syntax-Parsing einen weit hö-

[18] Eine weitergehende Diskussion der Möglichkeiten, Zeitangaben in RDF festzuhalten, erfolgt in Abschnitt 3.4. Im Verlauf dieser Arbeit wird dort, wo die Unterscheidung von Tripeln und Quadrupeln unerheblich ist, weiterhin «Tripel» geschrieben, um Formulierungen nicht unnötig komplex zu machen. Ein RDF-Tripel ist ja als die zu extrahierende Grundeinheit zu betrachten, ein Quadrupel ist eine Erweiterung eines Tripels.

[19] Mit Stand vom 31. Oktober 2012 enthält die englischsprachige Wikipedia 4136059 Artikel, die deutschsprachige als zweitgrößte von allen 1483949 Artikel (vgl. `http://stats.wikimedia.org/EN/Sitemap.htm`).

heren rechentechnischen Aufwand erfordert als das Auswerten von Infoboxen, wie es im DBpedia-Projekt geschieht, war vor Beginn der Arbeit nicht klar, ob mit der zur Verfügung stehenden Hardware ein sinnvoller Umgang mit der englischsprachigen Wikipedia in der zur Verfügung stehenden Zeit überhaupt möglich ist. Gleichzeitig ist die deutschsprachige Wikipedia jedoch so groß – die zweitgrößte aller Sprachvarianten –, dass auch aus ihr Tripel-Extraktionen in einem quantativ hohen Maße erfolgen können.

Zum anderen jedoch ist es als sehr sinnvoll anzusehen, WIKI2RDF zuerst für das Deutsche zu entwickeln und später für das Englische anzupassen, im Gegensatz zur umgekehrten Vorgehensweise: Da die deutsche Sprache in der Flexion komplexer und in der Wortstellung freier als das Englische ist, besteht bei einer Entwicklung für das Englische die Gefahr, WIKI2RDF soweit zu vereinfachen, dass es später einen hohen Aufwand bedeutet, es für andere indoeuropäische Sprachen anzupassen. Umgekehrt ist es jedoch einfach, ein Tool für das Deutsche später für das Englische (und für andere indoeuropäische Sprachen) anzupassen.[20]

3.1 Lemmatisierung, PoS-Tagging und *dependency parsing*: ParZu und TreeTagger

Ausgangspunkt zur Formulierung von Extraktionsregeln ist, wie erwähnt, eine lemmatisierte, mit PoS und Dependenz-Relationen versehene Form der Sätze aus den Volltexten.

Dies leistet WIKI2RDF nicht selbst, sondern ruft hierfür PARZU auf. PARZU[21] (Sennrich u. a., 2009) steht für «Zurich Dependency Parser for German» und ist ein in Prolog implementierter, unter der GNU General Public License zur Verfügung stehender *dependency parser* für das Deutsche. PARZU wurde in der vorliegenden Arbeit entsprechend den Anweisungen unter Linux installiert, wie empfohlen zusammen mit MORPHISTO[22] (Zielinski und Simon, 2008) zur morphologischen Analyse. Die Lemmatisierung und das PoS-Tagging leistet PARZU ebenfalls nicht selbst, sondern benötigt hierfür einen PoS-Tagger, dies war entsprechend der Empfehlung

[20] So ist, wie bereits erwähnt, für das Syntax-Parsing des Englischen nicht notwendigerweise eine Dependenz-Grammatik erforderlich, sondern es kann z. B. auch eine Kontextfreie Grammatik verwendet werden. Letztere ist jedoch für das Deutsche nicht geeignet. Umgekehrt können jedoch Dependenz-Grammatiken für beide Sprachen verwendet werden. Das bedeutet konkret, dass WIKI2RDF leicht für das Englische angepasst werden kann, sobald PARZU durch einen *dependency parser* für das Englische ersetzt wird. Solche Parser existieren.

[21] Beschreibung und Download unter `https://github.com/rsennrich/ParZu`. Online-Demo unter `http://kitt.cl.uzh.ch/kitt/parzu/`.

[22] `http://code.google.com/p/morphisto/`

```
1 Max Max N NE Masc|Nom|Sg 3 subj _ _
2 Mustermann Mustermann N NE _ 1 app _ _
3 studierte studieren V VVFIN 3|Sg|Past|_ 0 root _ _
4 von von PREP APPR Dat 3 pp _ _
5 1985 1985 CARD CARD _ 4 pn _ _
6 bis bis KON KON _ 5 kon _ _
7 1995 1995 CARD CARD _ 6 cj _ _
8 an an PREP APPR Dat 3 pp _ _
9 der der ART ART Def|Fem|Dat|Sg 10 det _ _
10 Universität Universität N NN Fem|Dat|Sg 8 pn _ _
11 Leipzig Leipzig N NE Neut|Dat|Sg 10 app _ _
12 . . $. $. _ 0 root _ _
```

Abbildung 3: Von ParZu erzeugte CoNLL-Ausgabe zum Beispielsatz

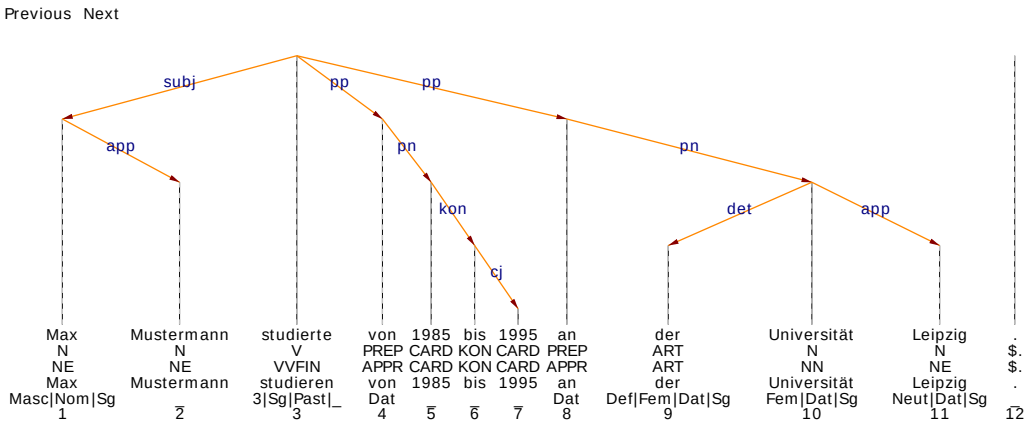

Abbildung 4: Von ParZu erzeugte SVG-Ausgabe zum Beispielsatz

TreeTagger[23] (Schmid, 1995, 1994). Es wurde das ParZu-Commit `7822e91` verwendet.

Die Funktionsweise von ParZu soll hier nicht näher beschrieben werden, wichtig für diese Arbeit sind die Ergebnisse: ParZu gibt standardmäßig Daten im CoNLL-Format[24] aus. Abb. 3 zeigt die Ausgabe für den Beispielsatz «Max Mustermann studierte von 1985 bis 1995 an der Universität Leipzig.», Abb. 4 zur Veranschaulichung eine graphische Darstellung dessen, die ParZu auf Wunsch ebenfalls erzeugen kann.

Der Beispielsatz hat 12 Token, wobei das letzte der Satzendepunkt ist. Im CoNLL-Format werden zu jedem Token 10 Felder gespeichert. Feld 1 enthält eine ID, Feld 2 das originale Wort, Feld 3 dessen Lemma, Feld 4 einen groben PoS-Tag, Feld 5 einen feineren PoS-Tag und Feld 6 morphologische Features. Die Felder 7 und 8 enthalten die Dependenz-Informationen, dabei gibt Feld 7 die ID des Kopfes an, Feld 8 die

[23] `http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/`
[24] `http://ilk.uvt.nl/conll/#dataformat`

Art der Relation. Die Felder 9 und 10 sind in der Standardeinstellung von ParZu unbesetzt und werden von wiki2rdf nicht ausgewertet.

Im Beispielsatz gibt es ein Wurzelelement 3 («studierte»), gekennzeichnet durch die Angabe «root» in Feld 8 sowie «0» in Feld 7. Direkte Kinder sind 1 («Max» als `subj`), 4 («von» als `pp`) und 8 («an» als `pp`). Die Kinder haben weitere Kinder, sodass die Ketten «Max Mustermann» (Subjekt), «von 1985 bis 1995» (Präpositionalphrase), «an Universität Leipzig» (Präpositionalphrase) sowie «an Universität der» (Präpositionalphrase) entstehen.

Die in Feld 8 verwendeten Labels werden in Kurzform in der ParZu beiliegenden Datei `LABELS.md` und ausführlicher in Foth (2004) erläutert.

wiki2rdf benötigt also von jedem zu bearbeitenden Wikipedia-Artikel also die ConLL-Ausgabe der geparsten Volltextabsätze sowie der Überschriften und Listenpunkte. Zudem müssen sämtliche Anker der verlinkten Entitäten (Outlinks) von ParZu geparst werden, damit diese in den ConLL-Ausgaben identifiziert werden können. Hierin besteht die *named entity recognition*.

3.2 Regeln

3.2.1 Regeln für die Extraktion aus Volltextabsätzen

wiki2rdf begreift Sätze als Ketten (*chains*), die von einem Kopf (*head*) ausgehen. Dabei unterscheidet es zwischen dem Subjekt sowie den Objekten. Innerhalb der Objekte wird nicht nach ihrer Natur bzw. Funktion unterschieden. Ketten entstehen, indem vom Kopf alle Kinder, deren Kinder usw. gesucht werden. Aus obigem Beispielsatz ergeben sich, wie beschrieben, also folgende Ketten, wobei Kette 1 das Subjekt ist:

1. Max Mustermann
2. von 1985 bis 1995
3. an Universität Leipzig
4. an Universität der

Man stelle sich vor, der Beispielsatz erscheint im Artikel zu Max Mustermann. Will man die Aussage extrahieren, dass Max Mustermann an der Universität Leipzig studiert hat, kann eine Regel hierfür also lauten:

```
<s=Max Mustermann> , <studieren> , <an> <o=Universität Leipzig> == <s> has_studied_at <o>
```

(Für dieses wie für alle folgenden Beispiele gilt, dass jede Regel komplett auf einer eigenen Zeile stehen muss. Aus Gründen des Layout werden sie in diesem Text jedoch umgebrochen, wenn sie zu lang sind. Solche Umbrüche sind durch eine Einrückung sowie das Zeichen ↪ am Beginn der nächsten Zeile gekennzeichnet.)

Die Regel zerfällt in den Teil vor «==» und den Teil danach. Der erste Teil enthält seinerseits drei durch `,` getrennte Teile. Diese sollen bedeuten: Suche nach dem Verb «studieren». Wenn gefunden, suche nach «Max Mustermann» in allen Ketten der Subjektposition. Wenn gefunden, suche nach «an Universität Leipzig» in allen Ketten der Positionen außer dem Subjekt. Wenn alles gefunden ist, dann gehe zum Teil nach «==». Dort ersetze `<s>` durch das, was im ersten Teil mit «`s=`» markiert ist, also «Max Mustermann». Dann ersetze `<o>` durch das, was im ersten Teil mit «`o=`» markiert ist, also «Universität Leipzig».[25] Es entsteht also das Tripel:

```
Max_Mustermann has_studied_at Universität_Leipzig
```

Um dies zu verallgemeinern, kann folgende Regel formuliert werden:

```
<s=[[art]]> , <studieren> , <an> <o=[[school]]> == <s> has_studied_at <o>
```

Hier wurden Platzhalter eingeführt, um folgende Extraktion zu ermöglichen: Steht beim Verb «studieren» in der Subjektposition der Artikelname und in einer anderen Position das Muster «an + *Schule*», dann bilde das genannte Tripel. Diese Regel kann auf sämtliche Artikel über natürliche Personen angewandt werden. Die Platzhalter bezeichnen Klassen von Entitäten und sind durch `[[` und `]]` eingeschlossen. `[[school]]` steht hierbei für Schulen, Hochschulen, Universitäten ... aller Art, `[[art]]` ist ein spezieller Platzhalter und steht für den Artikelnamen selbst.[26]

Wie nun können die verlinkten Artikel (Outlinks) automatisch klassifiziert werden? Hierfür wurden drei Möglichkeiten ins Auge gefasst: Verwendung der Klassen der DBpedia-Ontologie, Auswertung des ersten Satzes des Artikels, der diesen ja in

[25] Gemeint sind bei der Ersetzung natürlich nicht die Zeichenfolgen «Max Mustermann» und «Universität Leipzig», wie sie in den Ketten erscheinen, sondern die zugehörigen URI-Repräsentationen im DBpedia-Namespace. Diese ergeben sich, indem WIKI2RDF zu jedem Ankertext einer verlinkten Zeichenfolge speichert, zu welchem Wikipedia-Artikel der Link führt. Zum Beispiel verweist die Zeichenfolge «Universität Leipzig» auf den Artikel `Universität_Leipzig`. Hieraus ergibt sich auch die DBpedia-URI. Vom Artikelnamen selbst (`Max_Mustermann`) ist die URI natürlich ohnehin bekannt.

[26] Es ist wichtig, zwischen Klassen von Entitäten zu unterscheiden. Es genügt nicht, zu prüfen, ob die Zeichenfolge «an + *irgendeine Entität*» erscheint, denn so würde fälschlicherweise auch der Satz «Max Mustermann studierte an der Gemeinen Fichte die Evolution der Nadelbäume.» gefunden.

der Regel definiert,[27] sowie Auswertung der Kategorien des Artikels. Gegenwärtig ist in WIKI2RDF nur die letzte Option implementiert, die anderen könnten jedoch in Zukunft folgen.

Im Folgenden wird die Klassifizierung anhand der Kategorien erläutert:

Das Kategoriensystem der Wikipedia ermöglicht die Einordnung eines Artikels in eine oder mehrere Kategorien. So gehört der Artikel zur Universität Leipzig zu folgenden Kategorien:

- Wikipedia:Gesprochener Artikel
- Universität Leipzig
- Hochschule in Leipzig
- Bildung und Forschung in Leipzig
- Universität in Deutschland

In der Webansicht der Wikipedia erscheinen die Kategorien am Schluss jedes Artikels.

Kategorien gehören ihrerseits zu anderen Kategorien, wodurch eine hierarchische Struktur entsteht. Kategorienzugehörigkeiten können mithilfe des Tools CATGRAPH[28] visualisiert werden. Meyer (2010, S. 16 – 31) hat gezeigt, dass die Kategorien und Relationen jedoch desto unverständlicher und unbrauchbarer werden, je höher man in der Struktur blickt. Verlässlich sind allerdings die untersten Stufen, also die Kategorien, denen ein Artikel direkt zugeordnet ist. So enthält etwa die Kategorie «Universität in Deutschland» alle Artikel über Universitäten in Deutschland. Die Kategorien folgen zudem einem einheitlichen Muster in ihrer Formulierung: Es existieren analog z. B. die Kategorien «Universität in Estland» sowie «Universität in China». Für die gesamte deutschsprachige Wikipedia gilt, dass die direkten Kategorienzuordnungen ein sinnvolles Mittel darstellen, um Artikel zu klassifizieren.[29]

Neben der Regelsprache zur Tripelextraktion muss also eine weitere einfache Sprache eingeführt werden, um Artikel anhand ihrer Kategorien zu klassifizieren. Klassifizierungsregeln können so aussehen:

[27] So lautet der erste Satz des Artikels `Universität_Leipzig`: «Die Universität Leipzig (Alma mater lipsiensis) ist die größte Hochschule in Leipzig.» Hier kann nach erfolgtem PARZU-Parsing nach dem Muster «sein + Hochschule» gesucht werden, um festzustellen, dass die Universität Leipzig eine Hochschule ist.

[28] `http://toolserver.org/~dapete/catgraph/`

[29] Analog verwenden Suchanek u. a. (2007) das Kategoriensystem der englischsprachigen Wikipedia für ihre Ontologie YAGO und machen dabei ebenso ausschließlich Gebrauch von der untersten Stufe. So erkennen sie beispielsweise daraus, dass der Artikel einer Person in die Kategorie «1879 births» eingeordnet ist, dass die Person im Jahr 1879 geboren wurde.

```
Category:/^Schule\b/ -> school
Category:/^Hochschule\b/ -> school
Category:/^Universität\b/ -> school
Category:/^Fachhochschule\b/ -> school
[[school]] -> organization
```

Hier wird angegeben, dass Artikel, die in einer Kategorie sind, deren Name mit «Schule», «Hochschule», «Universität» oder «Fachhochschule» als ganzem Wort beginnt, zur Klasse `school` gehören. Die Syntax hierfür lautet:

```
Category:/regex/ -> class
```

Dabei ist `regex` ein regulärer Ausdruck in Perl-Syntax, gegen den mindestens ein Kategorienname des Artikels matchen muss. `class` ist die Klasse.[30]

In der letzten Zeile des obigen Beispiels wird schließlich angegeben, dass die Klasse `school` ihrerseits zur Klasse `organization` gehört, also alle Artikel, die eine Schule beschreiben, auch eine Körperschaft beschreiben.

Die Klassifizierungsregeln werden in einer Plain-Text-Datei gespeichert, die von WIKI2RDF geladen wird.

Neben der speziellen, immer vorhandenen Klasse `art`, die den Artikel selbst enthält, soll zudem die Klasse `entities` als immer vorhanden definiert werden, die alle Outlinks enthält.

Zurück zu den Extraktionsregeln:

Wir erweitern die Regel um einige mögliche Präpositionen und vereinfachen sie dahingehend, dass wir `art` direkt als Subjekt des Tripels ohne den Umweg über `s=` angeben. Dies ist sinnvoll, da der größte Teil aller Tripel, die aus einem Artikel extrahiert werden, diesen als Subjekt oder Objekt haben werden, und außerdem, da auch Sätze denkbar sind, in denen die Artikelentität gar nicht genannt wird. Es ergibt sich:

```
<[[art]]> , <studieren> , <(an|am|in|ans|ins)> <o=[[school]]> ==
  ↦ <art> has_studied_at <o>
```

Dabei sind die Präpositionen ebenfalls mit einem regulären Ausdruck in Perl-Syntax ausgedrückt.

Ein komplexeres Beispiel ist das Muster «X ist Professor an Universität Y». Hier sind in der Objektposition zwei Kettenmuster zu beachten: «Professor» sowie «an + `school`». Die Regel gilt nur als erfüllt, wenn beide Muster gefunden wurden.[31] Dafür wird das Zeichen \ eingeführt, das anzeigt, dass dort eine neue Kette beginnt:

[30] *Perl Compatible Regular Expressions*, also reguläre Ausdrücke in Perl-Syntax sollen hier nicht näher beschrieben werden. Zur Einführung sei auf Kvale (2000) sowie Friedl (2006) verwiesen.

[31] Oft wird ein solcher Satz von PARZU zwar so geparst, dass eine einzige Kette «Professor + an + `school`» entsteht, es ist jedoch nicht davon auszugehen, dass dies immer so geschieht: Beim

```
<[[art]]> , <sein>, <(Professor.*|.*professor.*)> \ <(an|am|in|ans|ins)>
  ↦ <o=[[school]]> == <art> is_professor_at <o>
```

Zusätzlich wurde der reguläre Ausdruck für «Professor» so formuliert, dass nicht nur «Professor», sondern auch «Professorin», «Universitätsprofessor» usw. gefunden werden.

Für viele extrahierte Tripel ist eine zeitliche Einordnung sinnvoll. Z. B. ist es sinnvoll, nicht nur zu wissen, dass eine Person an Universität Y studiert hat, sondern auch, von wann bis wann. Ist eine Zeitangabe gewünscht, soll dies angezeigt werden, indem die Extraktionsregel folgendermaßen erweitert wird:

```
<[[art]]> , <studieren> , <(an|am|in|ans|ins)> <o=[[school]]> ==
  ↦ <art> has_studied_at <o> && date
```

Die Hinzufügung `&& date` am Schluss bedeutet, dass WIKI2RDF versuchen soll, zu dem Tripel eine Zeitangabe zu extrahieren. Es ist naturgemäß nicht davon auszugehen, dass dies immer gelingt.

Zusammenfassend ergibt sich also folgende Syntax für Regeln zur Extraktion aus Volltextabsätzen:

```
subject-chain [ [ \ subject-chain ] ... ] , verb ,
  ↦ object-chain [ [ \ object-chain ] ... ] ==
  ↦ <s> p <o> [ && date ]
```

Dabei lautet die Syntax von *subject-chain* und *object-chain*:

```
<regex> [ <regex> ... ]
ggfs. mit Klassen innerhalb von [[ und ]]
ggfs. mit s= und o= für Subjekt und Objekt des Tripels
```

Auch `verb` kann einen regulären Ausdruck nach der Perl-Syntax enthalten.

Ist das Subjekt oder sind die Objekte des Satzes für die Regel unerheblich, muss als Platzhalter `<*>` für `subject-chain` bzw. `object-chain` geschrieben werden. Es wird festgelegt, dass `<*>` immer matcht, auch wenn überhaupt keine entsprechende Kette existiert.

Alle Klassifizierungsregeln werden in einer Plain-Text-Datei gespeichert, die von WIKI2RDF geladen wird.

Als Subjekt und Objekt des Tripels sind nur URIs aus dem DBpedia-Namespace, also praktisch Entitäten, zu denen ein Wikipedia-Artikel existiert, zugelassen. Prädikate des Tripels können verschiedenen Namespaces entstammen, letztere können

Satz «X ist Professor in Hamburg» wird «in Hamburg» immer direkt an «sein» angeschlossen. Deswegen ist es sicherer, zwei Ketten zu formulieren. Erfüllt wird die Regel auch dann, wenn in der Ausgabe von PARZU nur eine Kette existiert, denn auf diese passen dann ja beide Bedingungen: Sie enthält dann sowohl «Professor» als auch «an + `school`».

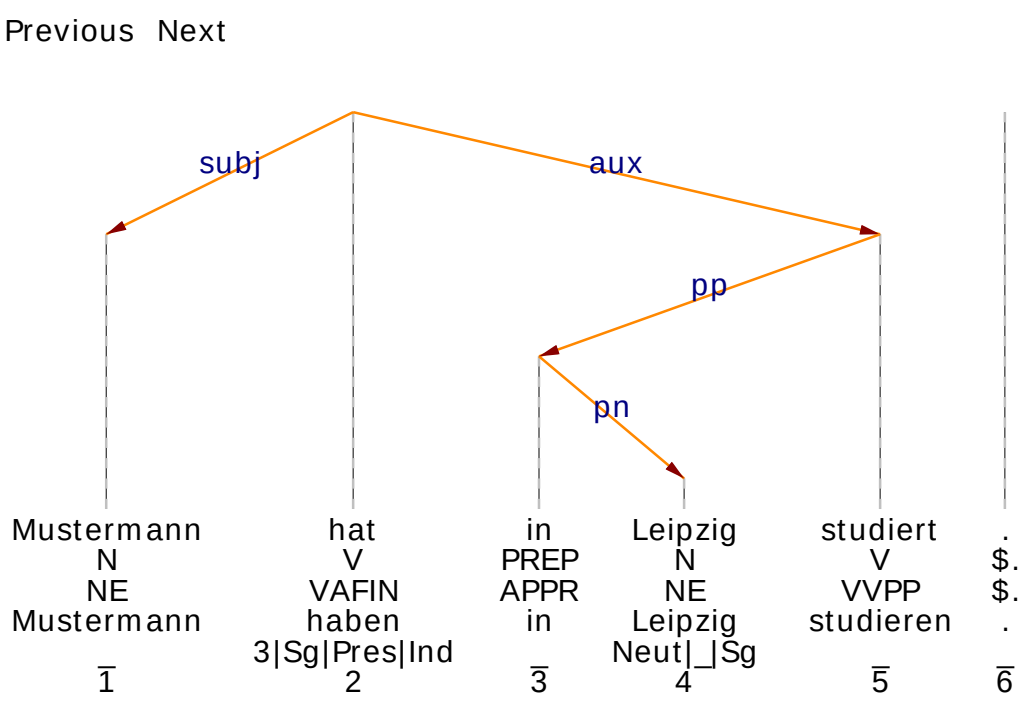

Abbildung 5: ParZu-Ausgabe des Satzes «Mustermann hat in Leipzig studiert.»

durch ein Präfix angegeben werden, das zu Beginn der Regeldatei definiert wird. Ist kein Präfix angegeben, wird ein wiki2rdf-eigener Namespace gewählt, derzeit vorübergehend ein nicht-dereferenzierbarer:

```
http://wiki2rdf.ibi.hu-berlin.de/property/
```

3.2.2 Besonderheiten im Parsing von Volltextabsätzen

Der vorgestellte Beispielsatz ist ein einfacher Hauptsatz und enthält ein Prädikat, das lediglich aus einem einzelnen Wort besteht. Sobald jedoch zusammengesetzte Verbformen – solche mit Hilfs- oder Modalverben –, Nebensätze oder Relativsätze hinzutreten, ergeben sich andere Parsing-Bäume, die gesondert behandelt werden müssen.

Die Abbildungen 5, 6, 7 und 8 zeigen Beispiele hierfür. Es soll die Information extrahiert werden, dass Mustermann in Leipzig studiert hat.

Abb. 5 zeigt einen Satz im Perfekt. Die Objekte sind nach wie vor Kinder des Vollverbs «studieren», das Subjekt ist jedoch Kind des Hilfsverbs.

Abb. 6 zeigt einen Relativsatz, in dem die gesuchte Information steckt. Die Objekte sind Kinder des Verbs, das Subjekt ist jedoch Kopf des Verbs.

Abb. 7 zeigt einen Satz mit Konjunktion, wobei die gesuchte Information im hinteren Teil ist. Auch hier sind die Objekte Kinder des gesuchten Verbs «studieren»,

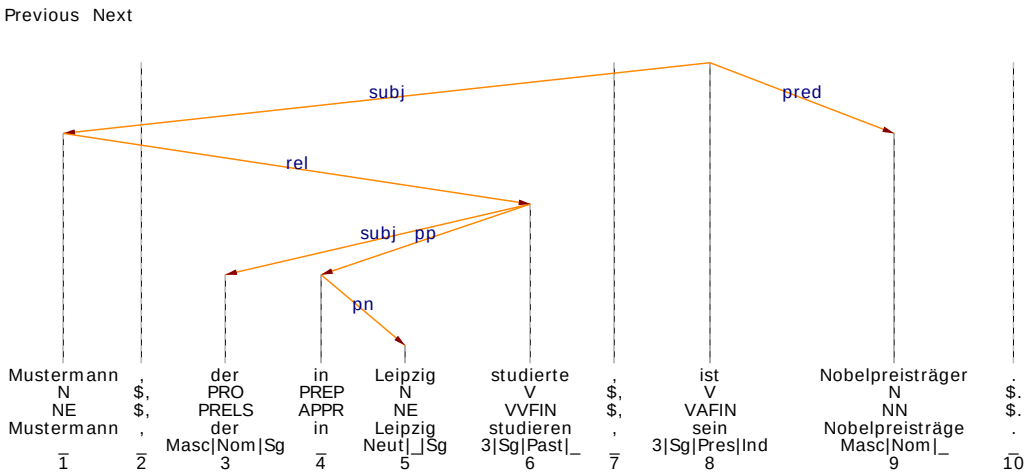

Abbildung 6: ParZu-Ausgabe des Satzes «Mustermann, der in Leipzig studierte, ist Nobelpreisträger.»

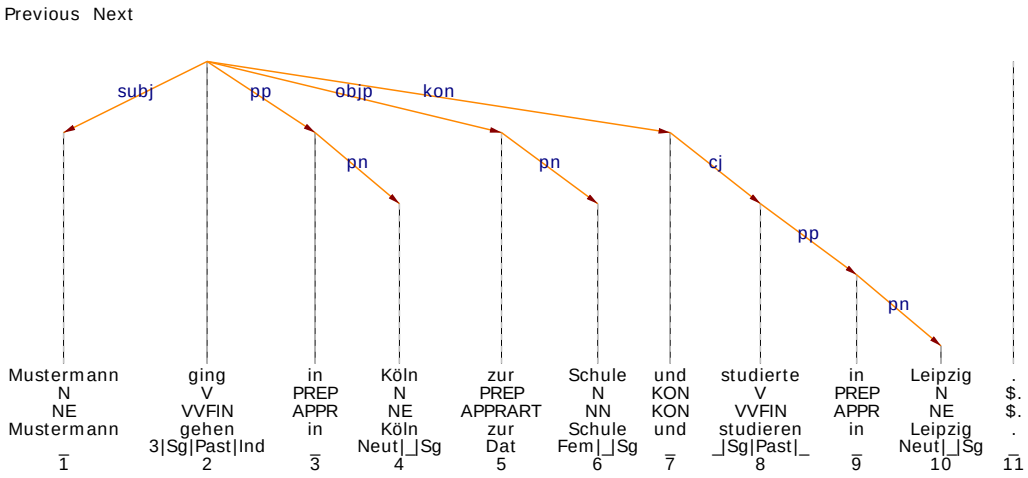

Abbildung 7: ParZu-Ausgabe des Satzes «Mustermann ging in Köln zur Schule und studierte in Leipzig.»

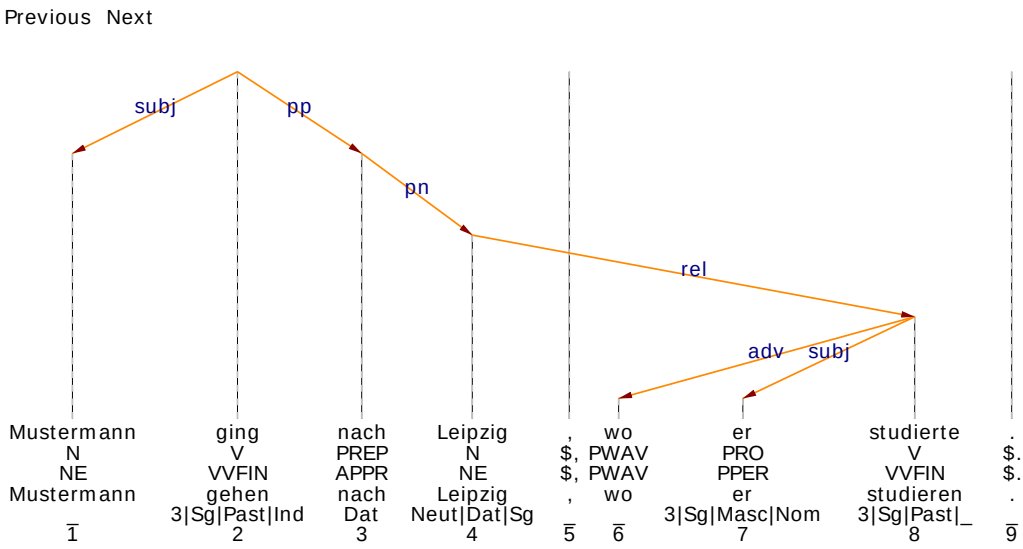

Abbildung 8: ParZu-Ausgabe des Satzes «Mustermann ging nach Leipzig, wo er studierte.»

das Subjekt ergibt sich jedoch, indem das Wurzelelement von «studieren» gesucht und davon das Kind `subj` gewählt wird.

Abb. 8 zeigt einen Relativsatz, in dem zwar das gesuchte Subjekt als `subj` zu «studieren» vorhanden ist – hier anaphorisch als «er», was im Rahmen der Anaphernresolution in Abschnitt 3.3 behandelt wird –, das Objekt jedoch als Kopf des Verbs und zudem ohne die Präposition «in».

Vorstellbar sind zudem weitere Konstruktionen sowie noch komplexere, etwa mit Relativsätzen im Perfekt.

Es ist unabdingbar, dass all diese Fälle anhand einer einzelnen Regel gefunden werden. Daher muss in wiki2rdf der Umgang mit verschiedensten Unterbringungsmöglichkeiten von Subjekt, Verb und Objekten implementiert werden.

Von den vorherigen Beispielen zu unterscheiden ist eine weitere zusammengesetzte Verbform im Deutschen: das Passiv. Abb. 9 zeigt hierfür einen Beispielsatz.

In diesem Beispiel existiert kein Agens, es ist jedoch eine Erweiterung um ein Agens vorstellbar: «X wurde von Y zum Professor ernannt.» Da dies auch als «Y ernannte X zum Professor.» formuliert werden kann, ist es sinnvoll, festzulegen, dass Extraktionsregeln immer im Aktiv formuliert werden und wiki2rdf bei Passivformen das grammatikalische Subjekt der Regel als «von + *Subjekt*» in der Objektposition sucht, sowie grammatikalische Objekte der Regel zudem in der Subjektposition.

Daraus ergibt sich, dass Formulierungen, die kein Agens haben bzw. bei denen das Agens unerheblich ist, dennoch aktivisch in einer Regel festgehalten werden müssen, mit `<*>` als Platzhalter im Subjekt. Um den Satz aus Abb. 9 zu finden, ist also

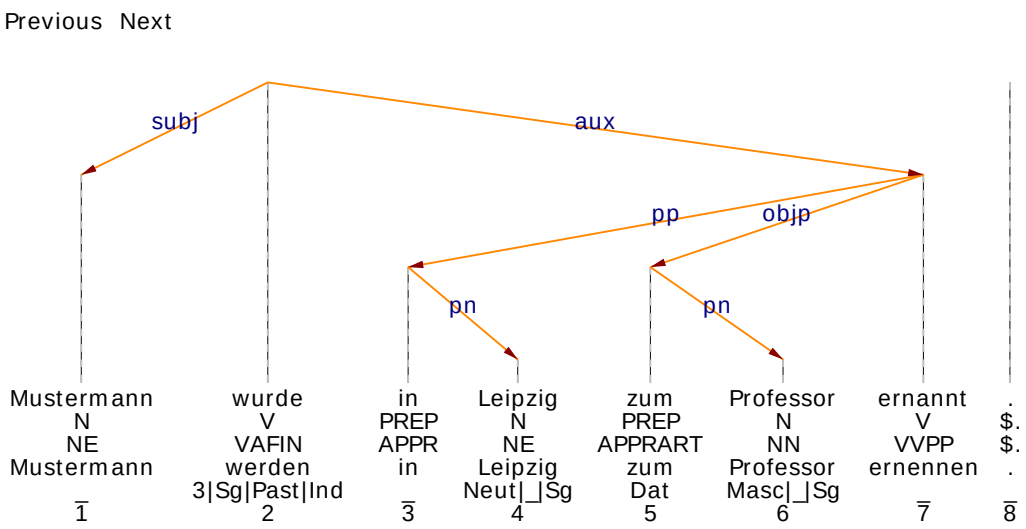

Abbildung 9: ParZu-Ausgabe des Satzes «Mustermann wurde in Leipzig zum Professor ernannt.»

folgende Regel erforderlich:

```
<*> , <ernennen> , <[[art]]> \ <zum> <Professor> \ <in> <o=[[ort]]> ==
  ↦ <art> is_professor_in <o> && date
```

3.2.3 Regeln für die Extraktion aus Listen

Wie erwähnt, können auch Listen im Artikelvolltext Informationen enthalten, die extrahiert werden sollen. Hier ist die Extraktion und damit auch die Regelsyntax einfacher. Regeln für Listen sollen mit `*` beginnen, in Anlehnung an Wikitext, in dem Aufzählungspunkte ebenfalls mit `*` beginnen.

Eine beispielhafte Regel für Artikel über natürliche Personen lautet:

```
* <o=[[award]]> , <(Ehrung.*|Auszeichnung.*|Preis.*)> ==
  ↦ <art> has_won_award <o> && date
```

Hier gliedert sich der linke Teil der Regel (vor `==`) in nur zwei Teile, getrennt durch `,`. Davon enthält der erste Teil nach dem `*` mindestens eine Kette, die im Listenpunkt enthalten sein muss, der zweite mindestens eine Kette, die in der Überschrift enthalten sein muss. Das Beispiel bedeutet also: Suche in einer Liste, die mit «Ehrung», «Auszeichnung» oder «Preis» überschrieben ist, nach Listenpunkten, die einen `award`, also einen Preis enthalten. Die extrahierte Aussage ist, dass die Person den Preis gewonnen hat, ggfs. mit Zeitangabe.

Verallgemeinernd haben Listenregeln also folgende Syntax:

```
list-chain [ [ \ list-chain ] ... ] ,
  ↳ heading-chain [ [ \ heading-chain ] ... ] ==
  ↳ <s> p <o> [ && date ]
```

`list-chain` und `heading-chain` haben die gleiche Form wie `subject-chain` und `object-chain` in Regeln für Absätze.

3.3 Anaphernresolution

WIKI2RDF benutzt derzeit einen einfachen, an Hobbs (1978) angelehnten Algorithmus zur Anaphernresolution. Er umfasst nur wenige Typen von Anaphern und ist ausbaufähig. Dennoch erzielt er bereits gute Resultate, da, wie dargestellt, die Anaphernresolution in Lexikonartikeln nicht so aufwendig ist wie in beliebigen Texten. Im Gegensatz zum Hobbs-Algorithmus ist er nicht auf Pronomina beschränkt.

Da je nach Klasse einer Entität Anaphern, die sich auf sie beziehen, unterschiedliche Formen haben können, ist die Anaphernresolution in WIKI2RDF abhängig von der Klasse der gesuchten Entität. Das bedeutet, dass ohne vorher geladene Klassifizierungsregeln (vgl. S. 21 dieser Arbeit) keine Anaphern aufgelöst werden.

Angaben zu Anaphern erfolgen im YAML-Format (Ben-Kiki u. a., 2009) in der Datei `coreferences.yaml`, die von WIKI2RDF geladen wird. In der Standardeinstellung enthält diese Datei folgende Angaben, die naturgemäß beliebig erweitert werden können:

```
male_person:
  - pronoun: 'er'
  - pronoun: 'dies'
  - pronoun: 'jen'
  - replacement: [ '.* ', '' ]
female_person:
  - pronoun: 'sie'
  - pronoun: 'dies'
  - pronoun: 'jen'
  - replacement: [ '.* ', '' ]
place:
  - anaphora: 'Stadt'
  - anaphora: 'Ort'
  - anaphora_with_preposition: 'dort'
school:
  - anaphora: 'Schule'
  - anaphora: 'Hochschule'
  - anaphora: 'Universität'
  - anaphora_with_preposition: 'dort'
```

Es werden, wie zu sehen ist, drei Typen von Anaphern unterschieden: `pronoun` (Pronomen), `anaphora` (nicht-pronominale Anaphern), `anaphora_with_preposition` (nicht-pronominale Anaphern, die einen Ausdruck mit Präposition ersetzen) sowie `replacement` (Anaphern, die gebildet werden, indem Teile des Antezedens-Ausdrucks durch andere ersetzt werden).

Die obigen YAML-Angaben legen folgende Anaphern fest:

- für Männer (Klasse `male_person`): Die Pronomen «er», «dieser» und «jener» – angegeben werden die Lemmata, also Feld 3 der ConLL-Ausgabe – sowie ein regulärer Ausdruck zur Ersetzung aller Vornamen durch den leeren String, sodass der Nachname übrig bleibt. Letzteres bedeutet, dass «Mustermann» als Anapher zu «Max Mustermann» erkannt wird.

- für Frauen (Klasse `female_person`): analog zu Männern, jedoch mit «sie» anstatt «er».

- für Orte (Klasse `place`): Die Anaphern «Stadt» und «Ort» sowie «dort» . Die Definitionen führen, wenn das jeweilige Antezedens «Leipzig» lautet, zu folgenden Auflösungen:

  - *Die Stadt* liegt in Sachsen. ⇒ *Leipzig* liegt in Sachsen.
  - *Der Ort* liegt in Sachsen. ⇒ *Leipzig* liegt in Sachsen.
  - Mustermann studierte *dort.* ⇒ Mustermann studierte *in Leipzig. (Präposition «in» wird hinzugefügt.)*

- für Bildungseinrichtungen (Klasse `school`): «Schule», «Hochschule», «Universität» und «dort».

Wird ein solcher anaphorischer Ausdruck gefunden, sucht WIKI2RDF in allen vorangegangen Sätzen nach möglichen Bezugswörtern. Zu allen möglichen Antezedenzien wird der Abstand (in Wörtern) zur Anapher bestimmt. Das mögliche Bezugswort mit dem geringsten Abstand wird als Antezedens bestimmt.

Die wahrscheinlich häufigsten Anaphern in jedem Wikipedia-Artikel sind solche, die sich auf das Artikellemma selbst beziehen. Dieses wird zwar automatisch in die Klasse `art` eingeordnet, zudem aber auch nach den Klassifizierungsregeln ausgewertet. Damit wird – sofern eine entsprechende Klassifizierungsregel existiert – ein Artikel, der eine männliche Person beschreibt, auch der Klasse `male_person` zugeordnet, sodass alle «er» nach der Person aufgelöst werden können. Der Algorithmus nimmt dabei bevorzugt an, dass sich Anaphern, die sich auf das Artikellemma beziehen könnten, tatsächlich auf dieses beziehen.

3.4 Zeitangaben

Wie bereits erwähnt, kann auf Wunsch zu einem Tripel ein zeitlicher Kontext, vulgo eine Zeitangabe, extrahiert werden.[32] Dies ist immer dort sinnvoll, wo das Tripel keine allgemeingültige Aussage beschreibt («Die Gemeine Fichte ist ein Baum.»), sondern eine Aussage mit zeitlichem Bezug: «Max Mustermann arbeitete in Köln», «Max Mustermann arbeitete in Leipzig» – wer diese Aussagen liest, möchte höchstwahrscheinlich auch wissen: Von wann bis wann jeweils?

Zeitangaben können also sowohl Zeitpunkte als auch Zeitspannen sein.

Die Extraktion von Zeitangaben aus Sätzen ist ebenfalls nicht trivial.[33] Derzeit funktioniert sie in WIKI2RDF nur für absolute Zeitangaben, sofern diese Kinder oder Kindeskinder des Verbs der Extraktionsregel sind bzw. – bei Listenregeln – sie im Listenpunkt stehen. Durch Mustererkennung werden so Zeitpunkte wie «am 1. August 1995» oder Zeitspannen wie «von März 1995 bis Juni 1996» erkannt.[34] Die Muster sind derzeit hartcodiert im Quelltext von WIKI2RDF und nicht durch externe Regeldateien änderbar.

Um eine umfassendere Erkennung von Zeitangaben zu gewährleisten, müssen die entsprechenden Funktionen in Zukunft erweitert werden. Dies betrifft mindestens:

- die Erkennung von relativen Zeitangaben: «1985 zog Mustermann nach Köln. *Zwei Jahre später* begann er dort ein Studium.» ⇒ Mustermann studierte ab 1987.

- die Erkennung von zeitlichen Kontexten, die im jeweiligen Satz überhaupt nicht explizit formuliert sind, sondern aus vorhergehenden Zeitangaben geschlossen werden müssen: «Von 1985 bis 1995 lebte Mustermann in Leipzig. Dort studierte er.» ⇒ Mustermann studierte zwischen 1985 und 1995 (wenn auch nicht notwendigerweise die gesamte Zeit).

Problematisch ist im Kontext von RDF zudem die Speicherung der Zeitangaben.

Damit sie eine einheitliche und international verständliche Form bekommen, werden sie zunächst nach den Regeln von ISO 8601 (ISO, 2004) umgewandelt, sodass aus «1. August 1995» `1995-08-01`, aus «1996» `1996` sowie aus der Zeitspanne «von Mai 1985 bis August 1995» `1985-05/1995-08` wird.

[32] Der Begriff «Zeitangabe» umfasst hier sowohl Datums- als auch Uhrzeitangaben. WIKI2RDF extrahiert jedoch derzeit nur Datumsangaben.

[33] Vgl. zur Einführung Jurafsky und Martin (2009, S. 777 – 786).

[34] Bei der Formulierung der Muster galt es auch, fehlerhafte Parsing-Bäume von PARZU zu umschiffen: Der Satz «Mustermann studierte bis 1995 Mathematik.» wird in der Regel so geparst, dass die Kette «bis Mathematik 1995» entsteht.

Doch wie kann diese Angabe im Tripel untergebracht werden? Hierfür gibt es nach wie vor keinen Standard und keine anerkannte best-practice. Als eleganteste Lösung erscheint die Erweiterung von Tripeln zu Quadrupeln, sodass *named graphs* entstehen.[35] Cyganiak u. a. (2012) schlagen eben diese Erweiterung von *N-Triples* zu *N-Quads* vor, wobei die vierte Position als *context* begriffen wird. Dies bezeichnet in der Realität oftmals die Provenienz des Tripels – z. B. publiziert DBpedia Quads, bei denen die vierte Position angibt, welcher Zeile des originalen Wikipedia-Artikels das Tripel entnommen wurde –, kann jedoch auch anderweitig besetzt werden: «The context element is also sometimes used to track a dimension such as time or geographic location.» (Cyganiak u. a., 2012, Abschn. 1)

Es wurde daher festgelegt, dass WIKI2RDF, wenn in der Extraktionsregel gewünscht, Quadrupel ausgibt, bei denen die letzte Position die Zeitangabe nach ISO 8601 enthält. Hierin unterscheidet sich bei Tripeln, die um *context* ergänzt werden, die Ausgabe von WIKI2RDF von den DBpedia-Datasets, bei denen *context* eben die Provenienz angibt.

Eine Zeitangabe nach ISO 8601 ist naturgemäß zunächst ein Literal. Cyganiak u. a. (2012) lassen als *context* auch Literale zu, in der Praxis hat sich jedoch gezeigt, dass die später zur Speicherung der Quadrupel eingesetzte Software 4STORE (vgl. Abschnitt 5.3) nur mit URIs als *context* umgehen kann. Aus diesem Grund müssen die Zeitangaben in URIs umgewandelt werden, es wird also ein URI-Namespace für Angaben nach ISO 8601 benötigt. Ein solcher Namespace mit dereferenzierbaren URIs wurde unabhängig voneinander bereits zweimal geschaffen[36], beide Namespaces sind jedoch unvollständig: Zum einen erlauben sie nur komplette Datumsangaben mit Jahr, Monat und Tag. In unserem Fall ist jedoch davon auszugehen, dass oftmals nur Jahresangaben erscheinen. Zum anderen können Zeitspannen in ihnen nur in der ISO-8601-Notation für Dauer (`P...`) angegeben werden, nicht in der ebenso standardisierten Notation mit Schrägstrich (*`Beginn/Ende`*), die hier benötigt wird. Zudem kommt keinem der beiden Namespaces ein Status als Quasi-Standard zu.[37]

Aus diesen Gründen wurde ein eigener Namespace für ISO-8601-Angaben erschaffen. Die URIs sind als temporär anzusehen und nicht dereferenzierbar, sie dienen zunächst nur dazu, die Zeitangaben aufzunehmen. Der Namespace ist:

[35] Möglich ist auch eine Lösung mittels *RDF reification* (Hayes, 2004, Abschn. 3.3.1), dies wird jedoch nicht empfohlen: «We discourage the use of RDF reification as the semantics of reification are unclear and as reified statements are rather cumbersome to query with the SPARQL query language.» (Bizer u. a., 2007, Abschn. 2.2) Vgl. auch Prud'hommeaux und Seaborne (2006, Abschn. 2.9) für ein Beispiel der Schwierigkeiten im Retrieval mit reifizierten RDF-Statements.

[36] vgl. `http://vocab.org/placetime/` sowie `http://wiki.linkedgov.org/index.php/RDF/Representing_dates`

[37] `http://vocab.org/placetime/` warnt zudem explizit: «This URI space is *experimental* and may be subject to change in the future. Please do not rely on the existence of any URIs in this space for production works.»

```
http://wiki2rdf.ibi.hu-berlin.de/datetime/
```

Beispielhaft ergeben sich also:

- `http://wiki2rdf.ibi.hu-berlin.de/datetime/1995-08-01`
- `http://wiki2rdf.ibi.hu-berlin.de/datetime/1996`
- `http://wiki2rdf.ibi.hu-berlin.de/datetime/1985-05/1995-08`

Ist eine Zeitangabe laut Extraktionsregel gewünscht, wird jedoch keine gefunden, gibt WIKI2RDF als *context* aus:

```
http://wiki2rdf.ibi.hu-berlin.de/datetime/null
```

Ist keine Zeitangabe gewünscht, gibt WIKI2RDF ein Tripel ohne *context* aus.

Obgleich WIKI2RDF also oftmals Quadrupel ausgibt, ist im Verlauf dieser Arbeit weiterhin von Tripeln die Rede, da zum einen WIKI2RDF, wenn gewünscht, nur Tripel erzeugt, zum anderen Quadrupel als Erweiterungen von Tripeln um eine vierte Stelle angesehen werden können. Ein Quadrupel enthält also ein Tripel.

3.5 Vorschlagsmodus

Es ist vorgesehen, dass Extraktionsregeln vom Anwender von WIKI2RDF manuell geschrieben werden. Dabei liegt auf der Hand, dass der Anwender nicht vorher wissen kann, welche Formulierungen in den Wikipedia-Artikeln überhaupt erscheinen und er also, auf sich allein gestellt, wahrscheinlich nicht alle erfasst.

Aus diesem Grund umfasst WIKI2RDF neben dem Extraktions- auch einen sogenannten Vorschlagsmodus (*suggest mode*). Hierbei gibt WIKI2RDF für die angegebenen Artikel Folgendes aus:

- alle Vorkommen von Verben mit dem Artikel im Subjekt und einer anderen verlinkten Entität im Objekt
- alle Vorkommen von Verben mit dem Artikel im Objekt und einer anderen verlinkten Entität im Subjekt (findet auch Passivkonstruktionen, die laut Extraktionsregelwerk ja aktivisch formuliert werden müssen)
- alle Vorkommen von Verben mit dem Artikel im Objekt, unabhängig vom Subjekt (findet u. a. Passivkonstruktionen ohne Agens)
- alle Aufzählungspunkte von Listen mit einer verlinkten Entität sowie irgendeiner Überschrift

Die Ausgabe erfolgt in der Syntax der Extraktionsregeln. Dies bedeutet, dass der Anwender mit der Ausgabe des Vorschlagsmodus bereits vorformulierte Regeln erhält, bei denen lediglich unnötige Ketten gelöscht bzw. zu spezifische Ketten verallgemeinert werden sowie das zu erstellende RDF-Tripel ergänzt werden müssen.

Es kann gewählt werden, ob im Vorschlagsmodus die konkreten Entitäten ausgegeben oder die Klassifizierungsregeln geladen und die Entitäten klassifiziert werden sollen. Letzteres bedeutet, dass z. B. anstelle der konkreten `Universität_Leipzig` aus dem Artikeltext bereits `[[school]]` ausgegeben wird. Existiert in den Klassifizierungsregeln keine Klasse für die Entität, so werden stattdessen deren Kategorien ausgegeben, damit der Anwender anhand derer eine neue Klassifizierungsregel formulieren kann.

Sinnvoll ist es, den Vorschlagsmodus über eine Stichprobe aus allen Artikeln, aus denen extrahiert werden soll, laufen zu lassen und die so erstellten Extraktionsregeln dann auf alle Artikel anzuwenden.

3.6 Desiderata

Obwohl WIKI2RDF schon viele Phänomene aus Wikipedia-Volltexten abdeckt und erfolgreich eingesetzt wurde (siehe Abschn. 5), bleiben verschiedene Punkte bislang unberücksichtigt. WIKI2RDF ist also noch als unvollständig anzusehen und trägt daher derzeit die Versionsnummer 0.01. Desiderata sind mindestens:

- Zulassen von Literalen als Objekte von Tripeln. Bislang können die Subjekte und Objekte von Tripeln ausschließlich URIs aus dem DBpedia-Namespace, also Entitäten, zu denen ein Wikipedia-Artikel existiert, sein. Sinnvoll – und in DBpedia bereits umgesetzt – sind jedoch Literale, z. B. numerische Werte, als Objekte: Berg Z hat die Höhe «350» (in Metern). Ebenso sollen URIs anderer Namespaces als Objekte zugelassen werden.

- Erkennen von Strukturen ohne Verb als Ausgangspunkt für Tripel: «*Nach seinem Studium in Leipzig* ging Mustermann nach Köln.» ⇒ Mustermann studierte in Leipzig.

- Daran anknüpfend das automatische Ableiten bestimmter Strukturen aus einer einzelnen Regel: Es ist sinnvoll, wenn anhand der Regel «`<[[art]]> , <studieren> , <in> <[[place]]>`» auch die Struktur «Studium in `place`» gefunden wird, ohne dass hierfür eine eigene Regel aufgestellt werden muss.

- Erkennen von Sätzen, die auf etwas vorher Genanntes Bezug nehmen, ohne dass explizit ein anaphorischer Ausruck erscheint: «Die Oscars wurden verge-

ben. Gewonnen hat X.» ⇒ X hat einen *Oscar* gewonnen, dies steht aber im zweiten Satz nicht explizit.

- Neben Volltextabsätzen und Listen auch das Extrahieren von Tripeln aus Tabellen (solchen im Artikeltext, nicht den Infoboxen) sowie ggfs. anderen, bislang unberücksichtigten, Teilen der Artikel.

- Erkennen relativer Zeitangaben (siehe oben).

- Erkennen von Negationen: Aus dem Satz «Mustermann studierte nicht in Leipzig.» würde derzeit extrahiert werden, *dass* Mustermann in Leipzig studiert hat. Es ist jedoch zu untersuchen, welche Rolle Negationen überhaupt in Lexikonartikeln spielen.

- Überprüfung, ob eine vorgeblich erkannte Entität wirklich diese ist: Man stelle sich den Satz «Der Bruder von Mustermann studierte in Leipzig.» vor. Wird «Mustermann» in der Subjektposition gesucht, würde dieser derzeit gefunden, obwohl nicht er, sondern sein Bruder gemeint ist. Es genügt hierbei nicht, lediglich zu überprüfen, ob die gesuchte Kette Kopf des Subjektes ist, denn in anderen Fällen kann es sinvoll sein, Tripel mit einem Subjekt auch dann zu extrahieren, wenn dieses nicht Kopf des grammatikalischen Subjekts ist: Sind bspw. an Niedersachsen grenzende Bundesländer gesucht, sollte auch der Satz «Der östliche Teil von Niedersachsen grenzt an Sachsen-Anhalt» herangezogen werden, um zu extrahieren, dass Niedersachsen an Sachsen-Anhalt grenzt, obwohl nicht «Niedersachsen», sondern «Teil» Kopf des Subjektes ist. Es müssen daher Regeln aufgestellt werden, welche Wörter außer der gesuchten Entität selbst als Kopf von Ketten zugelassen werden und welche nicht.

4 Umsetzung von wiki2rdf

Nachdem WIKI2RDF konzeptuell vorgestellt wurde, erläutert dieser Abschnitt konkret die Umsetzung und Bedienung. Er ist ohne Lektüre des vorangegangenen Abschnitts wahrscheinlich in Teilen unverständlich.

WIKI2RDF existiert derzeit in Version 0.01.[38] Es ist in Perl geschrieben und wurde mit Perl 5.12.4 unter Linux 2.6.24 getestet. WIKI2RDF erwartet in allen Belangen UTF8-Input (Kommandozeile, Dateien usw.) und gibt ausschließlich UTF8 aus.

Der Quelltext von WIKI2RDF ist auf der beiliegenden CD-ROM in der Datei `wiki2rdf.pl` enthalten. Auf einen Abdruck des Quelltextes in dieser Arbeit wird verzichtet, Abschnitt 4.3 beschreibt die Algorithmen und konkreten Funktionsnamen, jedoch ohne Programmcode.

4.1 Aufruf

Die Ausgabe von WIKI2RDF wird mithilfe verschiedener Kommandozeilenoptionen gesteuert. Übliche Aufrufe im Extraktionsmodus sind:

```
wiki2rdf.pl -r file -c file [-f format] article [article ...]
wiki2rdf.pl -r file -c file [-f format] -C category [category ...]
```

Dabei wird durch `-r` die Datei angegeben, aus der die Extraktionsregeln gelesen werden (*rule file*), durch `-c` die Datei, aus der die Klassifizierungsregeln gelesen werden (*classifier file*).

Standardmäßig können ein oder mehrere Wikipedia-Artikel angegeben werden, aus denen extrahiert werden soll. Ist die Option `-C` gesetzt, werden nicht Artikel, sondern eine oder mehrere Wikipedia-Kategorien angegeben. Dabei werden alle Artikel aus der Kategorie sowie ihrer Unterkategorien geladen. So werden durch die Angabe `-C Wissenschaftler` alle Artikel aus der Kategorie «Wissenschaftler» berücksichtigt, also alle Artikel über Wissenschaftler.

Leerzeichen in Artikel- und Kategoriennamen müssen mit Unterstrich (_) geschrieben werden, wie in ihren HTTP-URLs in Wikipedia bzw. URIs in DBpedia.

`-f format` bestimmt das Ausgabeformat. Dabei sind folgende Formate möglich:

- `nq`: Tripel bzw. Quadrupel mit vollständigen Namespaces. Beispiel:

[38] Die geringe Versionsnummer wurde gewählt, da WIKI2RDF für einen Produktiveinsatz noch nicht genügend getestet wurde, sowie aufgrund verschiedener Desiderata (siehe Abschnitt 3.6).

```
<http://de.dbpedia.org/resource/Ilse_Jahn>
  ↪ <http://wiki2rdf.ibi.hu-berlin.de/property/works_at>
  ↪ <http://de.dbpedia.org/resource/Deutsche_Akademie_der_Wissenschaften_zu_Berlin>
  ↪ <http://wiki2rdf.ibi.hu-berlin.de/datetime/1962/1967> .
```

Dies ist die Standardeinstellung, da auch die herunterladbaren DBpedia-Datasets Tripel mit vollständigen Namespaces enthalten.

- `nq_prefixes`: besser lesbare Ausgabe mit Präfixen. Beispiel:

```
@prefix prop: <http://wiki2rdf.ibi.hu-berlin.de/property/> .
@prefix db: <http://de.dbpedia.org/resource/> .
@prefix datetime: <http://wiki2rdf.ibi.hu-berlin.de/datetime/> .
db:Ilse_Jahn prop:works_at db:Deutsche_Akademie_der_Wissenschaften_zu_Berlin
  ↪ datetime:1962/1967 .
```

- `internal`: internes Format, nicht für den Produktionsbetrieb. Beispiel:

```
Ilse_Jahn works_at Deutsche_Akademie_der_Wissenschaften_zu_Berlin
  ↪ && 1962/1967
```

Der Vorschlagsmodus wird mit `--suggest` aufgerufen:

```
wiki2rdf.pl --suggest [-t] [-t] [-c file [--classify]] article [article ...]
wiki2rdf.pl --suggest [-t] [-t] [-c file [--classify]] -C category [category ...]
```

`-t` kann bis zu zweimal angegeben werden und bedeutet, dass die Ausgabe knapper oder noch knapper wird: dass von den vorgeschlagenen Regeln Teile gelöscht werden, die möglicherweise überflüssig sind und zu Unübersichtlichkeit führen. Bei einfacher Angabe sind dies Teile von Ketten, die nach einer erkannten Entität stehen, bei zweifacher Angabe sind es Ketten ohne erkannte Entität. Mindestens Letzteres führt jedoch dazu, dass unter Umständen wichtige Ketten nicht mehr erscheinen, zum Beispiel die Kette «Professor» in der Konstruktion «X ist Professor in Leipzig». Die Option `-t` erfordert also Vorsicht und Wissen um ihre Nachteile.

Eine Datei mit Klassifizierungsregeln muss auch im Vorschlagsmodus angegeben werden (`-c` *`file`*), wenn Anaphernresolution funktionieren soll. Darauf aufbauend kann die Option `--classify` gesetzt werden, wodurch in den ausgegebenen Regeln nicht konkrete Entitäten, sondern Klassen oder Wikipedia-Kategorien erscheinen (siehe S. 34 zur näheren Erläuterung).

Daneben gibt es sowohl im Extraktions- als auch im Vorschlagsmodus die Option `--sample` *`num`*. Sie gibt an, dass nicht alle angegebenen Artikel bzw. alle Artikel der angegebenen Kategorien ausgewertet werden sollen, sondern nur eine Stichprobe. Ist

$num >= 1$, wird es als die absolute Größe der Stichprobe verstanden. Ist $num < 1$, wird es als prozentuale Angabe der Größe der Stichprobe verstanden. Umfasst die angegebene Kategorie 1000 Artikel und ist $num = 0.01$, so werden also 10 Artikel (1 %) ausgewählt. Das gleiche geschieht, wenn $num = 10$ ist. Die Auswahl erfolgt zufällig.

Es bietet sich an, `--sample` in folgendem Workflow zu verwenden:

1. Vorschlagsmodus auf Zufallsstichprobe aus gewünschter Kategorie anwenden:

```
wiki2rdf.pl --suggest --sample 0.01 -c cfile --classify
 ↪ -C Naturwissenschaftler
```

2. Formulierung von Regeln anhand der Stichprobe
3. Schreiben der Regeln in die Datei `rfile`
4. Extraktion von Tripeln aus allen Artikel der Kategorie:

```
wiki2rdf.pl -r rfile -c cfile -C Naturwissenschaftler
```

Außerdem existiert folgender Modus:

```
wiki2rdf.pl --browse -C category [category ...]
```

`--browse` bedeutet, dass lediglich alle Artikel der Kategorien sowie ihrer Unterkategorien angezeigt werden. Es findet keine Extraktion statt. Dieser Aufruf ist sinnvoll, um einen Überblick über die Inhalte einer Kategorie zu gewinnen.

Zuletzt gibt es folgende allgemeine Optionen für alle Modi:

- `-v` / `--verbose`: ausführliche Ausgaben zu Kontroll- und Debugging-Zwecken (kann bis zu viermal angegeben werden, wobei die Ausgaben immer ausführlicher werden)
- `-h` / `--help`: lediglich Ausgabe einer Optionenübersicht
- `--version`: lediglich Ausgabe der Versionsnummer

4.2 Umgebung

WIKI2RDF muss in einem Verzeichnis mit folgenden Unterverzeichnissen laufen, die es – bis auf `dumps/` – selbst anlegt, wenn nötig:

- `dumps/`: Hier muss manuell der Abzug (Dump) der Wikipedia abgelegt werden, den WIKI2RDF benutzen soll. Dumps der deutschsprachigen Wikipedia können unter `http://dumps.wikimedia.org/dewiki/` heruntergeladen werden. Dabei gibt es verschiedene Dateien, die verschiedene Teile der Wikipedia abdecken. WIKI2RDF benötigt:
  - `dewiki-`*`Datum`*`-pages-articles.xml.bz2`: Alle Artikel in Wikitext mit XML als Wrapper
  - `dewiki-`*`Datum`*`-page.sql.gz`: SQL-Tabelle mit Angaben zu *pages*, das sind nicht nur Artikel, sondern auch Kategorien und andere Formate. Enthalten sind ID, Name usw.
  - `dewiki-`*`Datum`*`-categorylinks.sql.gz`: SQL-Tabelle mit Angaben zur Zugehörigkeit von Artikeln und Kategorien zu Kategorien
  - `dewiki-`*`Datum`*`-redirect.sql.gz`: SQL-Tabelle mit Angaben zu Redirects (Artikelweiterleitungen, vulgo Synonyme zum Artikellemma).
  - `dewiki-`*`Datum`*`-pagelinks.sql.gz`: SQL-Tabelle mit Angaben zu Outlinks von Artikeln (wird derzeit nicht benötigt, da WIKI2RDF die Angaben direkt dem Volltext der Artikel entnimmt)

  Das Verzeichnis `dumps/` ist das Einzige, das manuell befüllt werden muss. Alle folgenden Verzeichnisse werden von WIKI2RDF mit ihren Inhalten eingerichtet, falls noch nicht vorhanden.

- `bundles/`: Zum schnelleren Auffinden von Artikeln erzeugt WIKI2RDF aus der großen Datei `dewiki-`*`Datum`*`-pages-articles.xml.bz2` kleinere Bündel (*bundles*) von derzeit je 10000 Artikeln zusammen mit Listen darüber, welche Artikel sich in welchem *bundle* befinden. Die *bundle*s werden in diesem Verzeichnis abgelegt. Sie sind notwendig, da im Dump die Artikel nicht alphabetisch sortiert vorliegen, sondern vermutlich in der Reihenfolge, in der sie einmal angelegt wurden.

- `tables/`: Aus den SQL-Dumps `page`, `categorylinks`, `redirect` und `pagelinks` erzeugt WIKI2RDF kleinere Plain-Text-Dateien, genannt *tables*, die nur die wirklich benötigten Angaben enthalten. Diese werden mit GZIP komprimiert und hier abgelegt.

- `articles/`: Da auch das Heraussuchen eines Artikels aus einem *bundle* Zeit kostet, werden gefundene Artikel hier in je einer eigenen Datei abgelegt. Dabei wird der Wikitext so belassen und die Datei GZIP-komprimiert. Wird ein

einmal gelesener Artikel später wieder benötigt, muss er also nicht mehr aus einem *bundle* herausgesucht werden, sondern kann aus diesem Verzeichnis gelesen werden.

- `hashes/`: Hier wird in einer eigenen Datei pro Artikel die ConLL-Ausgabe von ParZu abgelegt. Dies ist sinnvoll, da auch das Parsen eines Artikels Zeit kostet und so ein einmal geparster Artikel später nicht noch einmal von ParZu bearbeitet werden muss. Die Bezeichnung *hash* für die hier abgelegten Dateien leitet sich nicht von einer Hashfunktion, sondern von der gleichnamigen Datenstruktur in Perl ab.[39] Die *hashes* werden im YAML-Format und gzip-komprimiert abgelegt.

4.3 Algorithmen

Dieser Abschnitt stellt die wichtigsten Algorithmen und Funktionen von wiki2rdf in knapper und vereinfachter Form als Pseudocode dar. Dabei wird nur der Extraktionsmodus behandelt. Namen von `Funktionen` und Variablen entsprechen dabei den realen im Perl-Quelltext und können dort näher betrachtet werden.

Die Ausgabe einzelner Funktionen kann zudem gut verfolgt werden, indem durch mehrfache Angabe der Option `-v` auf der Kommandozeile ausführliche Statusmeldungen angefordert werden.

Nach Aufruf von wiki2rdf müssen zunächst verschiedene Daten geladen werden:

```
create_bundles ;                                  // wenn bundles nicht existieren
load_tables ;                                        // tables laden bzw. erzeugen
if Artikel angegeben then
  | articles ← angegebene Artikelnamen
else if Kategorie angegeben then
  | foreach Kategorie do
  |   | articles ← articles + browse_down_category(Kategorie)
rules_raw ← Extraktionsregeln aus Datei;
classifiers ← Klassifizierungsregeln aus Datei;
coreferences ← Anapherregeln aus coreferences.yaml;
```

Dabei gibt die Funktion `browse_down_category` alle Artikel zurück, die zu einer Kategorie und deren Unterkategorien gehören.

[39] Genauer gesagt, ist es eine mehrstufige Kombination aus *hash* und *array*, in der die geparsten Artikel von wiki2rdf gespeichert werden.

Danach läuft folgender Hauptalgorithmus:

```
foreach article in articles do
    texts ← article_text(article);
    texthash ← article_to_hash(article);
    classes ← expand_classes(texthash(entities),Klassifizierungsregeln);
    add_coreferences(coreferences,classes);
    rule ← Perl-Array von expand_rules(rules_raw);
    Suche in texthash nach erfüllten Regeln (siehe S. 42);
```

`article_text` liefert den Wikitext eines Artikels, den die Funktion entweder aus einem *bundle* oder der entsprechenden Datei im Verzeichnis `articles/` liest. Wenn noch nicht vorhanden, schreibt sie den Wikitext nach `articles/`.

`article_to_hash` wandelt den Artikel in seinen sogenannten *hash* um (bzw. liest den *hash* aus `hashes/`, wenn schon vorhanden). Dies schließt den Aufruf von TreeTagger und ParZu ein:

```
if Datei in hashes/ existiert then
    hash ← Inhalt von Datei;
else
    Lösche Wikitext-Sonderzeichen (Links, Infoboxen ...) aus article;
    for_tree_tagger ← Absätze, Listen und Überschriften aus article;
    for_tree_tagger ← for_tree_tagger + Ankertexte der Outlinks aus article;
    for_tree_tagger ← for_tree_tagger + Lemma von article;
    Füge Lemma von article dem Lexikon von TreeTagger hinzu;
    Rufe TreeTagger mit for_tree_tagger auf;
    for_parzu ← Output von TreeTagger;
    Rufe ParZu mit for_parzu auf;
    hash ← Output von ParZu;
    // hash enthält geparste Absätze, Listen, Überschriften, Ankertexte der
       Outlinks sowie das Lemma
    Schreibe hash in Datei in hashes/;
return hash;
```

Funktion `article_to_hash(`*article*`)`

`expand_classes` klassifiziert die Outlinks in texthash nach den Klassifizierungsregeln, derzeit also anhand ihrer Kategorien. Die Funktion gibt einen *hash* (Perl-Datenstruktur) zurück, bei der jeder *key* eine Klasse ist und jeder Wert ein *array* der geparsten Outlinks in der Klasse.

`add_coreferences` erzeugt alle Anaphern pro Klasse, schreibt diese in einen *hash*, der texthash hinzugefügt wird.

`expand_rules` fügt denjenigen Regeln, in den Klassen referenziert werden, die geparsten Outlinks der jeweiligen Klassen hinzu. Dabei werden die Outlinks in Form eines regulären Ausdrucks eingesetzt, da das Pattern Matching in den Ketten ja anhand regulärer Ausdrücke geschieht.

Zur schnelleren Verarbeitung werden die Regeln schließlich in ihre Teile getrennt (an , und ==) und in ein *array* (Perl-Datenstruktur) geschrieben.

Der eigentliche Algorithmus zur Suche nach erfüllten Regeln in texthash ist:

```
foreach Zeile aus texthash do
    if Zeile ist Volltextabsatz then
        foreach w (Wort) do
            foreach rule (Textabsatzregeln) do
                if w entspricht Verb der Regel then
                    Suche Ketten entsprechend der grammatikalischen Konstruktion, in der
                    w steckt (siehe S. 43);
                    if rule erfüllt then
                        if Zeitangabe gewünscht then
                            suche Zeitangabe in Kindern von w
                        erzeuge Tripel;
    else if Zeile ist Listenpunkt then
        foreach rule (Listenregeln) do
            Suche Überschriftsketten laut Regel;
            Suche Listenpunktketten laut Regel;
            if rule erfüllt then
                if Zeitangabe gewünscht then
                    suche Zeitangabe in Kindern von w
                erzeuge Tripel;
```

Wie in Abschnitt 3.2.2 beschrieben, muss die Suche nach Ketten in Volltextabsätzen abhängig von der grammatikalischen Konstruktion gemacht werden, in der sich das Wort w befindet. Derzeit gilt hierfür folgender Algorithmus, der in Zukunft erweitert werden muss:

```
if w ist Vollverb in Haupt- oder Nebensatz then
    find_chains(Kinder subj von w, Subjekte der Regel);
    find_chains(andere Kinder von w, Objekte der Regel);
else if w ist adjektivisches Attribut then
    find_chains(Kopf von w, Subjekte der Regel);
    find_chains(Kinder von w, Objekte der Regel);
else if w ist in Relativsatz then
    if Relativpronomen ist Subjekt then
        find_chains(Kopf von w, Subjekte der Regel);
        find_chains(Kinder von w, Objekte der Regel);
    else
        find_chains(Kinder von w, Subjekte der Regel);
        find_chains(Kinder von w, Objekte der Regel);
        find_chains(Kopf von w, Objekte der Regel);
else if w ist in einer Passivform then
    find_chains(Kopf von w, Objekte der Regel);
    find_chains(Kinder von w, Objekte der Regel);
    find_chains(Kinder von w mit von oder durch, Subjekte der Regel);
else if w ist in einer anderen Form mit Hilfsverb (Perfekt usw.) then
    find_chains(Kopf von w, Subjekte der Regel);
    find_chains(Kinder von w, Objekte der Regel);
else if w ist Teil einer Konjunktionskette (X studierte A, ging nach B, arbeitete als C) then
    if subj ist lokal bei w then
        find_chains(Kinder subj von w, Subjekte der Regel);
        find_chains(andere Kinder von w, Objekte der Regel);
    else
        find_chains(Kopf von w, Subjekte der Regel);
        find_chains(Kinder von w, Objekte der Regel);
```

`find_chains` schließlich ist die Funktion, in der das eigentliche Pattern Matching stattfindet:

```
foreach Kette aus Regelteil do
    ruleregex ← regulärer Ausdruck von Kette aus Regelteil und Anaphern;
    foreach textword_chain aus allen Ketten, die von Start ausgehen do
        if textword_chain matcht ruleregex then
            if Match ist Anapher then
                possible_entities ← mögliche Antezedenzien;
                entity ← Antezedens aus possible_entities, das den geringsten Abstand zur
                Anapher hat;
            else
                entity ← Lemma zu Match
            matched_chains ← matched_chains +1;
            matched_entities ← matched_entities + entity;
return matched_chains,matched_entities;
```

Funktion `find_chains`(*Start,Regelteil*)

Das Extrahieren von Zeitangaben übernehmen die Funktionen `get_date` sowie `harmonize_dates`, die hier nicht näher erläutert werden sollen. Stehen mehrere Zeitangaben im Satz, werden solche bevorzugt, die direkt beim Verb der Regel stehen, da andere, die weiter weg sind – z. B. in Nebensätzen –, sich möglicherweise auf andere Aussagen beziehen.

5 Anwendung

Beispielhaft wurde WIKI2RDF angewandt, um Tripelextraktionen aus allen Artikeln der Wikipedia-Kategorie «Wissenschaftler» sowie deren Unterkategorien vorzunehmen. Diese umfassen im benutzten Dump vom 9. August 2012[40] insgesamt 68820 Artikel über Wissenschaftler.[41]

Es wurden eine Zufallsstichprobe von 100 Artikeln aus der Kategorie «Naturwissenschaftler» genommen und anhand der Ausgabe des Vorschlagsmodus 150 Regeln zur Extraktion formuliert. Das Vokabular umfasst 27 Prädikate.[42]

Die Regeln wurden daraufhin auf alle Artikel der Kategorie «Wissenschaftler» angewandt. Es wurden 244563 vollständige Tripel extrahiert.

Dieser Abschnitt stellte die Ergebnisse dar und diskutiert sie.

5.1 Regeln und Vokabular

Ziel dieses Versuchs war es, die Funktionsfähigkeit von WIKI2RDF zu überprüfen. Es war nicht das Ziel, ein fertiges Vokabular zur Beschreibung von Wissenschaftlern zu schaffen. Aus diesem Grund sind sämtliche Prädikate Ad-hoc-Kreationen im Namespace `http://wiki2rdf.ibi.hu-berlin.de/property/` und nicht in einer konsistenten Ontologie definiert. Die Schaffung eines Begriffssystems für den Produktiveinsatz – dies bedeutet wohl vor allen Dingen die Ergänzung der DBpedia-Ontologie um neue Prädikate – bleibt als Aufgabe für die Zukunft.

Tabelle 1 stellt das Vokabular dar. Spalte 1 zeigt das Prädikat und zudem Platzhalter für Subjekt und Objekt. Wie erwartet, haben die meisten Tripel das Artikellemma (`<art>`) als Subjekt. `<o>` steht für verlinkte Entitäten. Spalte 2 zeigt, aus welchen Klassen `<o>` stammen kann, also Range bzw. Domain des Prädikats. `<art>` ist naturgemäß immer eine Person. Spalte 3 enthält eine knappe Erläuterung. Spalte 4 gibt an, wie viele Extraktionsregeln aufgestellt wurden, die das jeweilige Tripel zum Ziel haben, mithin also, wie viele Formulierungen in den Stichprobenartikeln für die gleiche Aussage gefunden wurden. Spalte 5 gibt an, welche Prädikate aus bestehenden Ontologien dem dargestellten am ehesten entsprechen und also als An-

[40] `http://dumps.wikimedia.org/dewiki/20120809/`

[41] Der größte Teil der Artikel behandelt je einen Wissenschaftler (Instanz). Zudem enthält die Kategorie auch einige Lemmata wie «Nachwuchsforscher», «Privatgelehrter» oder «Wissenschaftler». Diese fallen jedoch nicht ins Gewicht.

[42] Da zu Beginn dieses Versuchs nicht klar war, ob die Extraktion aus der kompletten Kategorie «Wissenschaftler» in der zur Verfügung stehenden Zeit realisiert werden könnte, war es zunächst gedacht, sich auf die Kategorie «Naturwissenschaftler», einer Unterkategorie von «Wissenschaftler» zu beschränken. Daher wurden die Regeln anhand von Artikeln aus «Naturwissenschaftler» entwickelt. Die Extraktion wurde später jedoch auf «Wissenschaftler» ausgedehnt. Die Regeln sind nicht naturwissenschaftlerspezifisch, weswegen sie übernommen werden konnten.

knüpfungspunkte dienen können bei der Erstellung einer Ontologie. Der Namespace `dbp` steht dabei für die DBpedia-Ontologie.

Für alle Tripel wurde bestimmt, dass zu ihnen Zeitangaben extrahiert werden sollen.

Tabelle 2 erläutert die Klassen.

Leicht ist die Unfertigkeit des Vokabulars zu erkennen. So wird z. B. an manchen Stellen ein Sachverhalt durch mehrere Tripel beschrieben: Die Aussage, dass jemand Professor ist, ist in verschiedenen Prädikaten enthalten: `is_professor_at` (wo: welche Hochschule?), `is_professor_in` (wo: welche Stadt?) `is_professor_of` (Fach). Besser wäre hier eine Konstruktion `is_professor`, die die verschiedenen Aspekte vereint. Dies könnte im Kontext von RDF mit *blank nodes* realisiert werden.[43] Derzeit wird nicht explizit festgehalten, wenn bis zu drei Tripel über eine Professur zu einem einzigen Sachverhalt gehören. Dies kann nur geschlossen werden, wenn alle drei über dieselbe Datumsangabe verfügen. Wenn allerdings keine Datumsangabe extrahiert werden konnte, ist derzeit bei zwei oder mehr Professuren nicht zu unterscheiden, welches Fach zu welcher Professur und damit zu welcher Hochschule gehört.

Weiterhin zeigt sich, dass einige Tripel aus anderen geschlossen werden können und, sofern ein bestimmtes anderes Tripel schon existiert, nicht mehr extrahiert werden müssten. Wurde `has_studied_at` *`school`* schon extrahiert, müsste die Aussage `has_studied_in` *`place`* nicht mehr extrahiert werden, da der Studienort ja der Ort ist, an dem sich die Hochschule befindet. Diese Fälle werden derzeit jedoch nicht von WIKI2RDF behandelt.[44]

Die gesamten Extraktionsregeln sind in der Datei `rules-wissenschaftler.txt` auf der beiliegenden CD-ROM enthalten und in Anhang A ab S. 63 abgedruckt. Die Klassifizierungsregeln stehen in der Datei `classifiers-wissenschaftler.txt` sowie im Anhang B ab S. 67.

5.2 Ergebnisse

Die extrahierten vollständigen Tripel sind auf der beiliegenden CD-ROM in der Datei `wissenschaftler.nq` enthalten. Sie wurden mit der Option `--f nq` von WIKI2RDF erzeugt, enthalten also volle URIs und keine Präfixe. Dies wurde so gewählt, da auch die herunterladbaren DBpedia-Datasets keine Präfixschreibweise verwenden.

[43] *Blank nodes* werden zwar nicht empfohlen – «We discourage the use of blank nodes.» (Bizer u. a., 2007, Abschn. 2.2) –, sind aber die eleganteste Möglichkeit, Sachverhalte mit mehr als zwei Aspekten in RDF sinnvoll zu vereinen. Dies ist einer der Nachteile von RDF. Besser geeignet zur Repräsentation solcher Sachverhalte sind z. B. Frames.

[44] `has_studied_in` und die analogen Bildungen wurden eingeführt, da in manchen Artikeln keine Hochschule genannt wird, sondern nur die Aussage «X studierte in Köln» gemacht wird.

Tripel/Prädikat	Klassen von <o>	Bedeutung	Regeln	Überschneidungen
<art> has_founded <o>	organization	von Person gegründete Körperschaft	2	
<art> has_place_of_phd <o>	school, place	Ort der Dissertation	11	
<art> has_studied <o>	wissenschaft	Fach, das studiert wurde	7	
<art> has_studied_at <o>	school	Hochschule, an der studiert wurde	9	dbp:almaMater, dbp:school, dbp:university
<art> has_studied_in <o>	place	Ort, an dem studiert wurde	7	
<art> has_subject_of_phd <o>	wissenschaft	Fach der Dissertation	7	
<art> has_won_award <o>	award	wissenschaftliche o. ä. Auszeichnung, Preis	6	dbp:award
<art> is_editor_of <o>	journal	Herausgeber einer Zeitschrift	2	
<art> is_executive_of <o>	organization	leitende Funktion in Körperschaft	6	dbp:ceo, dbp:chairperson
<art> is_honorary_doctor_at <o>	school	Ehrendoktor an Hochschule	3	
<art> is_lecturer_at <o>	school	Dozent (außer Professor) an Hochschule	5	
<art> is_lecturer_in <o>	place	Dozent (außer Professor) in Ort	4	
<art> is_lecturer_of <o>	wissenschaft	Dozent (außer Professor) für Fach	3	
<art> is_member_of <o>	association	Mitgliedschaft in Verband, Verein o. Ä.	8	foaf:member
<art> is_professor_at <o>	school	Professor an Hochschule	4	
<art> is_professor_in <o>	place	Professor in Ort	4	
<art> is_professor_of <o>	wissenschaft	Professor für Fach	4	
<art> is_student_of <o>	person	Student / Schüler von anderer Person	8	
<art> lives_in <o>	place	wohnt in Ort	2	
<art> see_also <o>	entities	allgemeines «siehe auch»	1	skos:related
<art> was_ennobled_in <date>	date[a]	wurde geadelt	3	
<art> was_imprisoned <date>	date[a]	wurde verhaftet	1	
<art> was_imprisoned_in <o>	place	wurde verhaftet in Ort	1	
<art> works_at <o>	organization, school	arbeitet bei Körperschaft	15	dbp:employer
<art> works_in <o>	place	arbeitet in Ort	10	
<art> works_with <o>	person	arbeitet zusammen mit Person	12	
<o> is_student_of <art>	person	<art> ist Lehrer von anderer Person	2	
<o> named_after <art>	entities	Entität wurde nach der Person benannt	3	

[a] Datum nach ISO 8601 als Objekt. Dieses Feature wurde in dieser Arbeit nicht näher beschrieben und soll in Zukunft ersetzt werden, indem das Datum auch hier *context* wird und ein anderes Objekt erscheint.

Tabelle 1: Vokabular für Wissenschaftler

Name	Oberklasse	Beschreibung
academic_degree		akademischer Grad
association	organization	Verband, Verein ...
award		Preis, Auszeichnung, Orden ...
company	organization	Unternehmen
female_person	person	Frau
journal		Zeitschrift, Zeitung
male_person	person	Mann
museum	organization	Museum
observatory	organization	Observatorium, Sternwarte
organization		Körperschaft
peerage		Adelstitel
person		Person
place		Ort (geographisch): Stadt, Land, Region ...
school	organization	Hochschule: Universität, Fachhochschule ...
wissenschaft		Wissenschaft (im deutschen Sinne, also inklusive Geisteswissenschaften, daher nicht als science übersetzt)

Tabelle 2: Klassen zur Extraktion

Insgesamt wurden 257145 Tripel extrahiert, davon enthielten 12582 nicht aufgelöste Anaphern. Abzüglich dieser ergeben sich also 244563 vollständige Tripel.[45]

Tabelle 3 zeigt die absoluten und relativen Häufigkeiten der Prädikate.[46] Zudem gibt Spalte 3 an, zu wie vielen Artikeln das Prädikat gefunden wurde.[47] Wie zu sehen ist, kam es oft vor, dass mehrere Tripel mit demselben Prädikat und unterschiedlichen Objekten aus einem Artikel extrahiert wurden.

Angesichts der großen Artikelzahl kann keine endgültige Aussage darüber getroffen werden, ob die Häufigkeitsverteilung der Verteilung der jeweiligen Aussagen über alle Artikel entspricht, oder ob bestimmte Aussagen nicht extrahiert werden konnten, da sie in Formulierungen standen, die nicht durch die Regeln oder durch WIKI2RDF abgedeckt wurden. Doch legt ein Blick auf die Prädikate nahe, dass diejenigen mit hoher Häufigkeit auch den Typen von Aussagen entsprechen, die oft in Lexikonartikeln über Wissenschaftler erscheinen.

Tabelle 4 wertet die Extraktion der Zeitangaben aus. Es ist erstaunlich, dass trotz des relativ simplen Extraktionsalgorithmus zu etwa zwei Dritteln der Tripel Zeitangaben extrahiert werden konnten. Etwa 33 % der Tripel sind ohne zeitlichen *context*, aber es ist wahrscheinlich, dass darunter viele Fälle sind, bei denen in den Artikeln

[45] Nicht aufgelöste Anaphern wurden von WIKI2RDF markiert, indem als Objekt `unresolved_anaphora` ausgegeben wurde. Diese Tripel sind zwecks Debugging in der Datei `unresolved_anaphora.nq` auf der CD-ROM gespeichert. Sie werden in den folgenden statistischen Auswertungen jedoch nicht berücksichtigt.

[46] In dieser wie in allen folgenden Tabellen wird statt des deutschen Kommas zur Trennung von Dezimalstellen der englische Dezimalpunkt geschrieben, zudem sind die Werte auf drei Dezimalstellen gerundet.

[47] Dabei kann ein Artikel mehrfach gezählt sein, z. B. wenn `works_with` je einen Wissenschaftler als Subjekt sowie Objekt hat.

abs. Häufigkeit	rel. Häufigkeit	Artikel für Prädikat	Prädikat
25171	10.292	18475	has_studied_at
23377	9.559	16100	is_professor_at
20093	8.216	14251	works_at
18913	7.733	13300	works_in
17258	7.057	9570	has_won_award
15326	6.267	10658	lives_in
14384	5.882	9341	is_member_of
14238	5.822	9198	has_studied
12891	5.271	8888	is_executive_of
12661	5.177	8981	has_studied_in
11408	4.665	9454	has_place_of_phd
10537	4.309	8219	is_lecturer_at
9035	3.694	9080	works_with
8017	3.278	9027	is_student_of
6993	2.859	5274	is_professor_in
4071	1.665	3280	is_lecturer_in
3994	1.633	3113	is_professor_of
3921	1.603	2092	see_also
2960	1.21	2215	named_after
2496	1.021	1772	is_honorary_doctor_at
1763	0.721	1457	is_lecturer_of
1711	0.7	1426	has_founded
1201	0.491	1058	has_subject_of_phd
1062	0.434	845	is_editor_of
637	0.26	600	was_imprisoned
382	0.156	378	was_ennobled_in
63	0.026	62	was_imprisoned_in
244563	100.00		

Tabelle 3: Verteilung der Prädikate

abs. Häufigkeit	rel. Häufigkeit	
82237	33.626	Tripel ohne Zeitangabe
130550	53.381	Tripel mit Zeitpunkt
7521		davon spezifischer als Jahr[a]
30864	12.62	Tripel mit Zeitspanne
216		davon spezifischer als Jahr[a]
912	0.373	syntaktisch falsche Zeitangaben mit Buchstaben

[a] mit Angabe von Monat bzw. Tag zusätzlich zum Jahr

Tabelle 4: Häufigkeit von Zeitangaben in Tripeln

überhaupt keine Zeitangabe gemacht wird und daher kein Fehler des Algorithmus vorliegt. 912 Zeitangaben sind syntaktisch falsch, das heißt, sie entsprechen nicht ISO 8601, sondern enthalten daneben noch andere Zeichen aus dem Artikeltext. Dies ist auf Fehler im Algorithmus zurückzuführen, die leicht verbessert werden können.

Nun soll ein Blick darauf geworfen werden, wie viele Tripel pro Artikel extrahiert wurden.[48] Abb. 10 zeigt ein Histogramm mit absoluten Häufigkeiten für Tripel pro Artikel.[49] Leider wird deutlich, dass die maximale Häufigkeit mit 12133 Artikeln (17.63 %) bei 0 Tripeln liegt, dass also zu 12133 Artikeln überhaupt kein Tripel extrahiert wurde. Danach fällt die Kurve ab. Aus 10113 Artikeln (14.695 %) wurden je 1 Tripel extrahiert, aus 9453 Artikeln (13.736 %) je 2 Tripel, aus 8400 Artikeln (12.206 %) je 3 Tripel. Insgesamt entfallen auf 1 bis 10 Tripel 76.566 % der Artikel. Zusammen mit den Artikeln mit 0 Tripeln ergibt dies also 94.196 % der Artikel zwischen 0 und 10 Tripeln. Das Histogramm in Abb. 11 wirft einen genaueren Blick auf dieses Intervall mit relativen Häufigkeiten. Anhang C enthält die vollständige Tabelle, die den Histogrammen zugrunde liegt.

Nach diesen reinen Häufigkeitsauswertungen drängt es, Recall (wie viele Aussagen, die extrahiert werden sollten, wurden extrahiert?) und Precision (wie viele der Tripel enthalten wahre Aussagen?) zu betrachten.[50] Der Recall ist jedoch derzeit nicht endgültig zu bestimmen, daher soll auf eine Berechnung an dieser Stelle verzichtet werden. Der Grund ist, dass in dem Ad-hoc-Vokabular die einzelnen Prädikate nur anhand von Beispielen definiert sind, nämlich durch die Regeln für ihre Extraktion. Etwa ist das Prädikat `has_studied_in` derzeit nur indirekt und unvollständig

[48] Auch in den folgenden Darstellungen ist ein Tripel mehreren Artikeln zugeordnet, wenn sowohl Subjekt als auch Objekt aus der Kategorie «Wissenschaftler» sind (vgl. Fußnote 47 auf S. 48).

[49] Zwei Ausreißer mussten manuell entfernt werden: Der Artikel `Max-Planck-Forschungspreis` mit 106 Tripeln sowie `Thomas-Körner-Preis` mit 170 Tripeln. Beides sind jedoch Artikel, die nicht in die Kategorie «Wissenschaftler» gehören sollten, da sie keine Wissenschaftler beschreiben.

[50] «Wahre Aussage» bedeutet hierbei, dass die Aussage so extrahiert wurde, wie sie im Artikel steht. Es bedeutet nicht notwendigerweise, dass die Aussage, die im Artikel gemacht wird, stimmt.

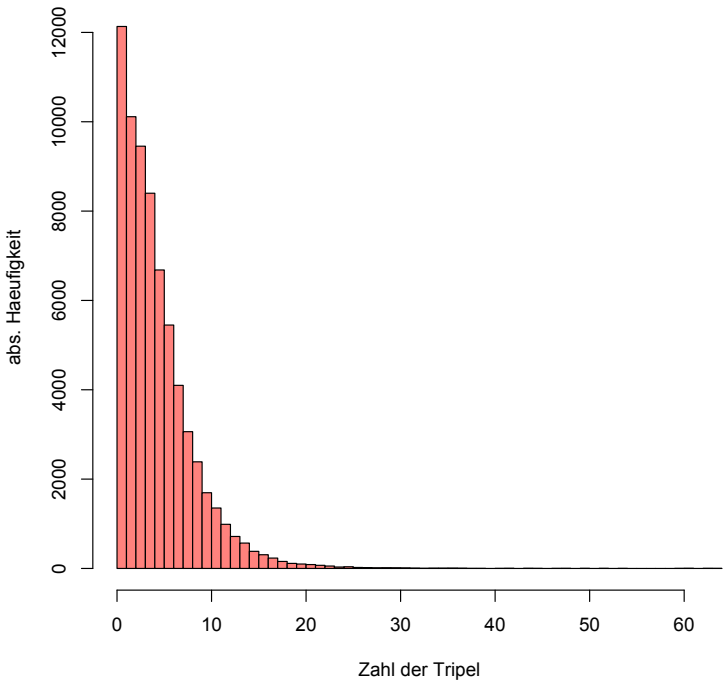

Abbildung 10: Histogramm der Tripel pro Artikel (komplett, abs. Häufigkeiten)

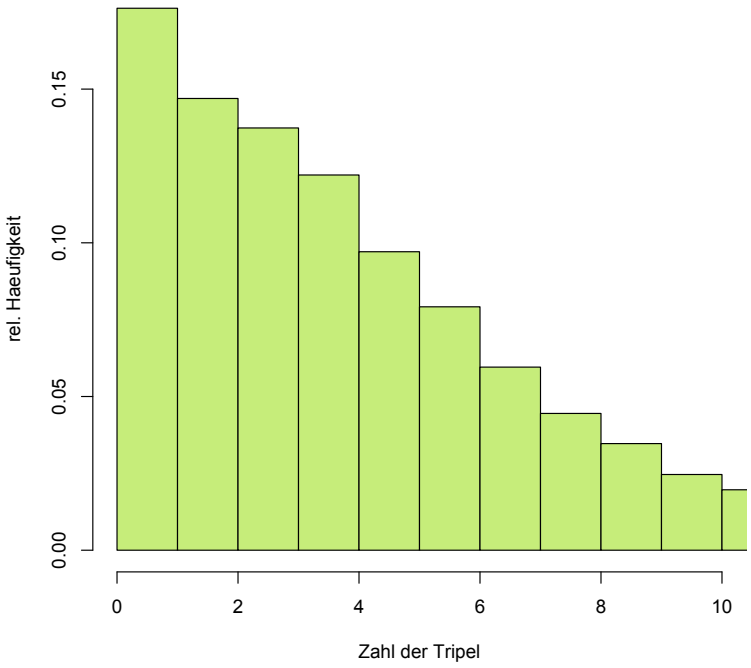

Abbildung 11: Histogramm der Tripel pro Artikel (0 bis 10 Tripel, rel. Häufigkeiten)

Tripel (ohne Berücksichtigung der Zeitangaben)	
korrekt	186
nicht korrekt	14
davon:	
wegen falsch aufgelöster Anapher	*3*
andere Gründe[a]	*11*
Summe	200

Zeitangaben (bei den 186 korrekten Tripeln)	
korrekt extrahiert	108
davon:	
unvollständig[b]	*8*
inkorrekte Zeitangabe extrahiert	7
entspricht nicht ISO 8601	4
keine Zeitangabe extrahiert	67
davon:	
weil im Artikel nicht gegeben	*45*
weil Zeitangabe relativ, daher nicht aufgelöst	*10*
andere Gründe[a]	*12*
Summe	186

[a] z. B.: von WIKI2RDF falsch ausgewertete Formulierungen in den Ketten, von PARZU falsch geparste Sätze

[b] z. B.: Es wurde nur ein Zeitpunkt extrahiert, obwohl im Satz eine Zeitspanne steht. Dennoch ist der extrahierte Zeitpunkt nicht falsch.

Tabelle 5: Analyse einer Zufallsstichprobe von 200 Artikeln

durch einige Beispiele für Formulierungen definiert. Umgekehrt ist jedoch nicht bestimmt, welche Aussagen *insgesamt* unter `has_studied_in` fallen. Es liegt also keine vollständige Definition vor. Daher kann nicht bestimmt werden, welche konkreten Aussagen noch hätten extrahiert werden sollen, aber nicht extrahiert wurden. Es ist aber angesichts noch fehlender Features von WIKI2RDF (siehe Abschn. 3.6) sowie möglicher Parsingfehler von PARZU davon auszugehen, dass der Recall noch erhöht werden kann und muss.

Die Precision jedoch kann anhand einer repräsentativen Stichprobe bestimmt werden. Dafür wurden 200 Tripel zufällig ausgewählt und manuell anhand der zugrunde liegenden Artikel ihre Richtigkeit geprüft. Es ergaben sich die Werte in Tabelle 5. Die 200 ausgewählten Tripel sind im Anhang D abgedruckt.

Betrachtet man die Tripelaussagen ohne ihren zeitlichen *context*, ergibt sich eine Precision von 93 % (186 wahre Aussagen). Fügt man die Zeitangaben hinzu, ergibt sich folgendes Bild: 108 Tripel haben korrekte Zeitangaben, sind daher vollständig

korrekt. Zu 45 Tripeln wurde im Ausgangsartikel überhaupt keine Zeitangabe gemacht, daher sind auch diese als vollständig korrekt zu werten. Zusammen ergeben sich also $108 + 45 = 153$ vollständig korrekte Tripel und daher eine endgültige Precision von 76.5 % in der Zufallsstichprobe.

Ein endgültiger Precision-Wert von 76.5 % ist als gut anzusehen angesichts des einfachen Extraktionsalgorithmus für Zeitangaben. Der Precision-Wert von 93 % ohne Berücksichtigung der Zeitangaben ist als sehr erstaunlich und sehr positiv zu werten!

Abschließend muss, um zu bewerten, was die Extraktionen an neuen Aussagen gebracht haben, noch bestimmt werden, wie viele der Aussagen in der DBpedia Deutsch bereits vorhanden waren. Dafür wurden Tripel aus den Datasets der DBpedia Deutsch gesucht, die entweder je dasselbe Subjekt und Objekt haben, wie die 200 in der Stichprobe, oder umgekehrt unser Subjekt als Objekt sowie unser Objekt als Subjekt. Zu ihnen wurde das Prädikat betrachtet. Es konnte in lediglich 2 Fällen festgestellt werden, dass Subjekt und Objekt durch ein Prädikat miteinander verbunden waren, das die Aussage einschließt, die von WIKI2RDF extrahiert wurde. Es ergibt sich also eine Überschneidung mit DBpedia Deutsch von 1 %.

5.3 Suchbeispiele: SPARQL-Abfragen

Im Folgenden werden einige beispielhafte Suchanfragen gezeigt, die mit den neu extrahierten Tripeln in Verbindung mit Tripeln aus DBpedia Deutsch möglich werden: Abfragen, die an die Wikipedia sowie an die DBpedia allein so vorher nicht gestellt werden konnten.

Dafür wurde mithilfe der Software 4STORE[51] ein *triple store* aufgesetzt, in den die extrahierten Tripel mit ihren *context*s sowie die benötigten Tripel aus der deutschsprachigen DBpedia geladen wurden. 4STORE wurde gewählt, da es, im Gegensatz zu den meisten anderen RDF-Datenbanken, N-Quads unterstützt. Leider erlaubt die eingesetzte Version 1.1.3 nur N-Quads im ASCII-Zeichensatz, also ohne spezifische UTF8-Zeichen.[52] Deswegen mussten die Tripel leider mithilfe des Perl-Moduls `Text::Unidecode`[53] in ASCII umgewandelt werden, wodurch die Sonderzeichen verloren gingen und die URIs im DBpedia-Namespace also nicht mehr den realen entsprechen. Dies ist zu beachten, wenn die Abfragen und Suchergebnisse betrachtet werden.

[51] `http://4store.org/`

[52] Tripel im Turtle-Format hingegen können von 4STORE auch dann geladen werden, wenn sie UTF8-Zeichen enthalten.

[53] `http://search.cpan.org/perldoc/Text::Unidecode`

Es wurden die folgenden SPARQL-Abfragen[54] gestellt. Die Suchergebnisse sind, sofern zahlenmäßig möglich, in Anhang E aufgeführt.

Es gelten die folgenden Präfixe:

```
PREFIX db: <http://de.dbpedia.org/resource/>
PREFIX prop: <http://wiki2rdf.ibi.hu-berlin.de/property/>
PREFIX dbprop: <http://dbpedia.org/ontology/>
PREFIX dbdeprop: <http://de.dbpedia.org/property/>
PREFIX datetime: <http://wiki2rdf.ibi.hu-berlin.de/datetime/>
PREFIX dcterms: <http://purl.org/dc/terms/>
```

1. Personen, die in Köln studiert haben, sortiert nach Datum:

```
SELECT DISTINCT ?s ?g
WHERE {
  {
    GRAPH ?g {
      ?s prop:has_studied_in db:Koln .
    }
  }
  UNION
  {
    GRAPH ?g {
      ?s prop:has_studied_at ?o .
    }
    ?o dbprop:locationCity db:Koln .
  }
}
ORDER BY ASC(?g)
```

   Ergebnis: 239 Treffer.

2. Ehrendoktoren der Humboldt-Universität zu Berlin, sortiert nach Datum der Verleihung der Ehrendoktorwürde:

```
SELECT DISTINCT ?s ?g
WHERE {
  GRAPH ?g {
    ?s prop:is_honorary_doctor_at db:Humboldt-Universitat_zu_Berlin .
  }
}
ORDER BY ASC(?g)
```

   Ergebnis: 30 Treffer.

[54] Zur Abfragesprache SPARQL für RDF vgl. Prud'hommeaux und Seaborne (2008).

3. Deutsche, die in den USA Professoren sind:

```
SELECT DISTINCT ?s ?g
WHERE {
  {
    GRAPH ?g { ?s prop:is_professor_at ?o . }
    ?s dcterms:subject <http://de.dbpedia.org/resource/Kategorie:Deutscher> .
    ?o dbdeprop:ort db:Vereinigte_Staaten .
  }
  UNION
  {
    GRAPH ?g { ?s prop:is_professor_at ?o . }
    ?s dcterms:subject <http://de.dbpedia.org/resource/Kategorie:Deutscher> .
    ?o dbdeprop:staat db:Vereinigte_Staaten .
  }
  UNION
  {
    GRAPH ?g { ?s prop:is_professor_in db:Vereinigte_Staaten . }
    ?s dcterms:subject <http://de.dbpedia.org/resource/Kategorie:Deutscher> .
  }
}
ORDER BY ASC(?g)
```

Ergebnis: 307 Treffer.

4. Personen, die zwischen 1920 und 1929 in Hannover aktiv waren.[55]

```
SELECT DISTINCT ?s ?p ?o ?stadt ?g
WHERE {
  {
    GRAPH ?g { ?s ?p ?stadt . }
  }
  UNION
  {
    GRAPH ?g { ?s ?p ?o . }
    ?o dbprop:locationCity ?stadt .
  }
  FILTER (?stadt=db:Hannover)
  FILTER regex(str(?g), "192.")
}
ORDER BY ASC(?g)
```

Ergebnis: 24 Treffer.

[55] Das Filtern nach Zeitraum erfolgt anhand des regulären Ausdrucks «`192.`» und umfasst derzeit keine Zeitspannen, die vor 1920 begannen und nach 1929 endeten und also nicht die Zeichenkette `192` enthalten.

5. Personen, die an derselben Hochschule studiert haben, an der sie später Professoren wurden.

```
SELECT DISTINCT ?s ?stud
WHERE {
  ?s prop:has_studied_at ?stud .
  ?s prop:is_professor_at ?prof .
  FILTER (?stud = ?prof)
}
```

Ergebnis: 1469 Treffer.

6. Personen, die an einer Hochschule mit bis zu 5000 Studenten studiert haben und Professoren wurden an einer Hochschule ab 30000 Studenten.[56]

```
SELECT DISTINCT ?s ?stud ?prof
WHERE {
  ?s prop:has_studied_at ?stud .
  ?s prop:is_professor_at ?prof .
  ?stud dbdeprop:studentenzahl ?studnum .
  ?prof dbdeprop:studentenzahl ?pstudnum .
  FILTER (?studnum<=5000)
  FILTER (?pstudnum>=30000)
}
```

Ergebnis: 44 Treffer.

7. deutsch-französische Zusammenarbeit:

```
SELECT DISTINCT ?a ?b
WHERE {
  {
    ?a prop:works_with ?b .
    ?a dcterms:subject <http://de.dbpedia.org/resource/Kategorie:Deutscher> .
    ?b dcterms:subject <http://de.dbpedia.org/resource/Kategorie:Franzose> .
  }
  UNION
  {
    ?a prop:works_with ?b .
    ?a dcterms:subject <http://de.dbpedia.org/resource/Kategorie:Franzose> .
    ?b dcterms:subject <http://de.dbpedia.org/resource/Kategorie:Deutscher> .
  }
}
```

Ergebnis: 44 Treffer.

[56] Die Ergebnisse sind mit Vorsicht zu genießen, da die Extraktion von Studentenzahlen in der DBpedia Deutsch derzeit nicht immer verlässliche Resultate erzeugt hat.

6 Schluss

Die exemplarische Anwendung von WIKI2RDF auf die Artikel der Kategorie «Wissenschaftler» der deutschsprachigen Wikipedia hat Ergebnisse erzielt, die in quantitativer Hinsicht beeindruckend sind und in qualitativer Hinsicht mindestens als sehr zufriedenstellend bezeichnet werden können. Letzteres gilt insbesondere angesichts der noch nicht sehr umfangreichen Algorithmen zur Anaphernresolution und zur Extraktion von Zeitangaben. Es ist davon auszugehen, dass durch ihre Erweiterungen sowie die Implementierung weiterer sprachlicher Konstrukte noch bessere Ergebnisse erzielt werden können.

Die jetzigen Ergebnisse zeigen jedoch zweierlei:

- Artikel der Wikipedia erlauben als Lexikonartikel aufgrund ihrer sprachlichen Struktur erfolgreiche Informationsextraktion mit relativ geringem Aufwand im Vergleich zu anderen Textsorten.
- Das regelbasierte Konzept von WIKI2RDF erweist sich als geeignet zur Tripelextraktion aus den Artikeln.

Es liegt nun – neben der Schaffung eines konsistenten Vokabulars für die Prädikate der bereits extrahierten Tripel – vor allen Dingen nahe, weitere Extraktionsregeln für andere Themenbereiche zu schreiben und also weitere Extraktionen zu ermöglichen. So kann DBpedia in hohem Maße erweitert werden, und zwar so, dass wirklich die *ganze* Wikipedia als Basis der Tripel zur Verfügung steht.

Daneben kann darüber nachgedacht werden, WIKI2RDF so zu erweitern, dass es nicht nur mit Wikitexten, sondern auch mit anderen Formaten umgehen kann. Dies ist vor allen Dingen dort leicht möglich, wo die *named entity recognition* nicht sehr aufwendig ist, z. B. in biographischen Texten aus anderen Quellen. So ist eine Software vorstellbar, die aus jeglichen Personenbiographien Aussagen extrahiert und in RDF-Tripel umwandelt. So könnten Personenbiographien verschiedener Quellen in ein Semantic-Web-Format gebracht werden, auch wenn diese noch nicht in strukturierter Form vorliegen. Dies könnte etwa Projekten wie dem «Personendaten-Repositorium»[57] (Roeder und Körner, 2011) helfen, das zum Ziel hat, personenbiographische Angaben verschiedener Quellen zusammenzubringen und unter einer einheitlichen Oberfläche verfügbar zu machen.

[57] `http://pdr.bbaw.de/`

Literatur

Auf sämtliche Internetquellen wurde zum letzten Mal am 3. Dezember 2012 zugegriffen.

Atserias, J., H. Zaragoza, M. Ciaramita und G. Attardi (2008). Semantically Annotated Snapshot of the English Wikipedia. In: N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odjik, S. Piperidis, und D. Tapias (Hrsg.), *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC '08)*, Marrakesch, Marokko, S. 2313 – 2316. European Language Resources Association (ELRA). `http://www.lrec-conf.org/proceedings/lrec2008/pdf/581_paper.pdf`.

Auer, S. und J. Lehmann (2007). What have Innsbruck and Leipzig in common? Extracting Semantics from Wiki Content. In: E. Franconi, M. Kifer, und W. May (Hrsg.), *The Semantic Web: Research and Applications, 4th European Semantic Web Conference, ESWC 2007, Innsbruck, Austria, June 3 – 7, 2007, Proceedings*, Volume 4519 of *Lecture Notes in Computer Science*, S. 503 – 517. Springer. Preprint: `http://www.informatik.uni-leipzig.de/~auer/publication/ExtractingSemantics.pdf`.

Ben-Kiki, O., C. Evans und I. döt Net (2009). *YAML Ain't Markup Language (YAML™) Version 1.2* (3. Aufl.). `http://yaml.org/spec/1.2/spec.pdf`.

Bizer, C. (2012). DBpedia 3.8 released, including enlarged Ontology and additional localized Versions. `http://blog.dbpedia.org/2012/08/06/dbpedia-38-released-including-enlarged-ontology-and-additional-localized-versions/`.

Bizer, C., R. Cyganiak und T. Heath (2007). How to Publish Linked Data on the Web. `http://sites.wiwiss.fu-berlin.de/suhl/bizer/pub/LinkedDataTutorial/20070727/`.

Bizer, C., J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak und S. Hellmann (2009). DBpedia. A Crystallization Point for the Web of Data. *Journal of Web Semantics 7*, S. 27 – 47. Preprint: `http://www.wiwiss.fu-berlin.de/en/institute/pwo/bizer/research/publications/Bizer-etal-DBpedia-CrystallizationPoint-JWS-Preprint.pdf`.

Buitelaar, P., P. Cimiano, M. Grobelnik und M. Sintek (2005). Ontology Learning from Text. Tutorial at ECML/PKDD 2005. Porto, Portugal, 3rd October – 7th Oc-

tober, 2005. In conjunction with the Workshop on Knowledge Discovery and Ontologies (KDO-2005). http://people.aifb.kit.edu/pci/OL_Tutorial_ECML_PKDD_05/.

Cyganiak, R., A. Harth und A. Hogan (2012). N-Quads. Extending N-Triples with Context. Public draft. http://sw.deri.org/2008/07/n-quads/.

Foth, K. A. (2004). *Eine umfassende Constraint-Dependenz-Grammatik des Deutschen.* Universität Hamburg. http://nats-www.informatik.uni-hamburg.de/pub/CDG/DeutschGrammar/doc.ps.

Friedl, J. E. F. (2006). *Mastering Regular Expressions* (3. Aufl.). Sebastopol: O'Reilly.

Gödert, W., K. Lepsky und M. Nagelschmidt (2012). *Informationserschließung und Automatisches Indexieren. Ein Lehr- und Arbeitsbuch.* X.media.press. Berlin: Springer.

Hayes, P. (Hrsg.) (2004). *RDF Semantics.* W3C. W3C Recommendation. http://www.w3.org/TR/rdf-mt/.

Herbelot, A. und A. Copestake (2006). Acquiring Ontological Relationships from Wikipedia Using RMRS. In: *Proceedings of the Workshop on Web Content Mining with Human Language Technologies, 2006, ISWC'06.* http://www.cl.cam.ac.uk/~ah433/athens.pdf.

Hitzler, P., M. Krötzsch, S. Rudolph und Y. Sure (2008). *Semantic Web. Grundlagen.* eXamen.press. Berlin: Springer.

Hobbs, J. R. (1978). Resolving pronoun references. *Lingua 44*, S. 311 – 338.

ISO (Hrsg.) (2004). *ISO 8601. Data elements and interchange formats. Information interchange. Representation of dates and times.* ISO.

Jentzsch, A. (2009). DBpedia. Extracting structured data from Wikipedia. Präsentation bei Semantic Web In Bibliotheken (SWIB2009), Köln, November 2009. http://www.anjajentzsch.de/slides/SWIB09_DBpedia.pdf.

Jurafsky, D. und J. H. Martin (2009). *Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition.* (2. Aufl.). Upper Saddle River: Pearson.

Krötzsch, M., D. Vrandecic, M. Völkel, H. Haller und R. Studer (2007). Semantic Wikipedia. *Journal of Web Semantics 5*, S. 251 – 261.

Kuhlen, R. (1977). *Experimentelle Morphologie in der Informationswissenschaft.* München: Verlag Dokumentation.

Kvale, M. (2000). Perl regular expressions tutorial. In: *Perl Programming Documentation.* Abrufbar mit `perldoc perltut` auf der Kommandozeile sowie unter `http://perldoc.perl.org/perlretut.html`.

Meyer, A. (2010). Begriffsrelationen im Kategoriensystem der Wikipedia. Entwicklung eines Relationeninventars zur kollaborativen Anwendung. Bachelorarbeit, Fachhochschule Köln. `http://d-nb.info/1009711954/34/`.

Nakayama, K. (2008). Wikipedia Mining for Triple Extraction Enhanced by Coreference Resolution. In: J. Breslin, U. Bojārs, A. Passant, und S. Fernández (Hrsg.), *Proceedings of the ISWC2008 Workshop on Social Data on the Web (SDoW2008), Karlsruhe, Germany, October 27, 2008.* `http://CEUR-WS.org/Vol-405/paper6.pdf`.

Nakayama, K., M. Pei, M. Erdmann, M. Ito, M. Shirakawa, T. Hara und S. Nishio (2008). Wikipedia Mining. Wikipedia as a Corpus for Knowledge Extraction. In: *Wikimania 2008*, Alexandria, Ägypten. `http://sigwp.org/en/images/0/06/Wikimania2008.pdf`.

Nguyen, D. P. T., Y. Matsuo und M. Ishizuka (2007). Relation Extraction from Wikipedia Using Subtree Mining. In: *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22 – 26, 2007, Vancouver, British Columbia, Canada*, S. 1414 – 1420. AAAI Press. `http://www.miv.t.u-tokyo.ac.jp/papers/dat-AAAI07.pdf`.

Nohr, H. (2005). *Grundlagen der automatischen Indexierung. Ein Lehrbuch* (3. Aufl.). Berlin: Logos.

Prud'hommeaux, E. und A. Seaborne (Hrsg.) (2006). *SPARQL Query Language for RDF.* W3C. Live draft. `http://www.w3.org/2001/sw/DataAccess/rq23/`.

Prud'hommeaux, E. und A. Seaborne (Hrsg.) (2008). *SPARQL Query Language for RDF.* W3C. W3C Recommendation. `http://www.w3.org/TR/rdf-sparql-query/`.

Roeder, T. und F. Körner (2011). The Person Data Repository. In: B. Maegaard (Hrsg.), *Supporting Digital Humanities. Copenhagen 17 – 18 November 2011. Conference Proceedings.* `http://crdo.up.univ-aix.fr/SLDRdata/doc/show/copenhagen/SDH-2011/submissions/sdh2011_submission_24.pdf`.

Schmid, H. (1994). Probabilistic Part-of-Speech Tagging Using Decision Trees. In: *Proceedings of International Conference on New Methods in Language Processing*, Manchester.

Schmid, H. (1995). Improvements in Part-of-Speech Tagging with an Application to German. In: *Proceedings of the ACL SIGDAT Workshop*, Dublin, Irland.

Sennrich, R., G. Schneider, M. Volk und M. Warin (2009). A New Hybrid Dependency Parser for German. In: C. Chiarcos, R. E. de Castilho, und M. Stede (Hrsg.), *Von der Form zur Bedeutung: Texte automatisch verarbeiten. From Form to Meaning: Processing Texts Automatically. Proceedings of the Biennial GSCL Conference 2009*, Tübingen, S. 115 – 124. Narr. `https://files.ifi.uzh.ch/cl/volk/papers/Sennrich_Schneider_Volk_Warin_Pro3Gres_GSCL.pdf`.

Suchanek, F. M., G. Kasneci und G. Weikum (2007). YAGO. A core of Semantic Knowledge unifying WordNet and Wikipedia. In: *WWW2007. Proceedings of the 16th international conference on World Wide Web*, New York, S. 697 – 706. `http://www2007.org/papers/paper391.pdf`.

Zielinski, A. und C. Simon (2008). Morphisto. An Open-source Morphological Analyzer for German. In: J. Piskorski, B. W. Watson, und A. Yli-Jyrä (Hrsg.), *Finite-State Methods and Natural Language Processing, 7th International Workshop, FSMNLP 2008, Ispra, Italy, September 11 – 12, 2008. Post-proceedings*, Volume 19 of *Frontiers in Artificial Intelligence and Applications*, S. 177 – 182. IOS Press.

Anhang

A Extraktionsregeln `rules-wissenschaftler.txt`

```
<[[art]]> , <gründen> , <o=[[organization]]> == <art> has_founded <o> && date
<[[art]]> , <sein> , <(Gründer.*|.*gründer.*)> \ <o=[[organization]]> == <art> has_founded <o> && date
<[[art]]> , <(beschreiben|schreiben)> , <in> <(Dissertation|Doktorarbeit)> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_place_of_phd <o> && date
<[[art]]> , <(erwerben|erlangen|erhalten|erreichen|machen|sein)> , <(Doktor.*|Ph. D.)> \ <(an|am|in|ans|ins|von)> <o=[[school]]> == <art> has_place_of_phd <o> && date
<[[art]]> , <(erwerben|erlangen|erhalten|erreichen|machen|sein)> , <(Doktor.*|Ph. D.)> \ <in> <o=[[ort]]> == <art> has_place_of_phd <o> && date
<[[art]]> , <promovieren> , <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_place_of_phd <o> && date
<[[art]]> , <promovieren> , <in> <o=[[ort]]> == <art> has_place_of_phd <o> && date
<[[art]]> , <(sein|erhalten)> , <(Doktorand.*|Promovend.*)> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_place_of_phd <o> && date
<[[art]]> , <(sein|erhalten)> , <(Doktorand.*|Promovend.*)> \ <in> <o=[[ort]]> == <art> has_place_of_phd <o> && date
<[[art]]> , <(verfassen|schreiben|beginnen|abschließen)> , <(Dissertation|Doktorarbeit)> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_place_of_phd <o> && date
<[[art]]> , <(verfassen|schreiben|beginnen|abschließen)> , <(Dissertation|Doktorarbeit)> \ <in> <o=[[ort]]> == <art> has_place_of_phd <o> && date
<*> , <promovieren> , <[[art]]> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_place_of_phd <o> && date
<*> , <promovieren> , <[[art]]> \ <in> <o=[[ort]]> == <art> has_place_of_phd <o> && date
<[[art]]> , <(beginnen|aufnehmen|abschließen|betreiben|absolvieren)> , <(Studium|.*studium)> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_studied_at <o> && date
<[[art]]> , <besuchen> , <o=[[school]]> == <art> has_studied_at <o> && date
<[[art]]> , <einschreiben> , <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_studied_at <o> && date
<[[art]]> , <(erwerben|erlangen|erhalten|erreichen|machen)> , <(Abschluss|Abschluß|Bachelor|Master|Diplom|.*diplom|Magister|A. B.|M. A.|M. S.|B. A.|B. S.)> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_studied_at <o> && date
<[[art]]> , <(erwerben|erlangen|erhalten|erreichen|machen)> , <[[academic_degree]]> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_studied_at <o> && date
<[[art]]> , <studieren>, <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_studied_at <o> && date
<[[art]]> , <(zuwenden|zuwandte)> , <Studium> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_studied_at <o> && date
<*> , <ausbilden> , <[[art]]> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_studied_at <o> && date
<*> , <einschreiben> , <[[art]]> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_studied_at <o> && date
<[[art]]> , <(beginnen|aufnehmen|abschließen|betreiben|absolvieren)> , <(Studium|.*studium)> \ <in> <o=[[ort]]> == <art> has_studied_in <o> && date
<[[art]]> , <(erwerben|erlangen|erhalten|erreichen|machen)> , <(Abschluss|Abschluß|Bachelor|Master|Diplom|.*diplom|Magister|A. B.|M. A.|M. S.|B. A.|B. S.)> \ <in> <o=[[ort]]> == <art> has_studied_in <o> && date
<[[art]]> , <(erwerben|erlangen|erhalten|erreichen|machen)> , <[[academic_degree]]> \ <in> <o=[[ort]]> == <art> has_studied_in <o> && date
<[[art]]> , <studieren>, <in> <o=[[ort]]> == <art> has_studied_in <o> && date
<[[art]]> , <(zuwenden|zuwandte)> , <Studium> \ <in> <o=[[ort]]> == <art> has_studied_in <o> && date
<*> , <ausbilden> , <[[art]]> \ <in> <o=[[ort]]> == <art> has_studied_in <o> && date
<*> , <einschreiben> , <[[art]]> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> has_studied_in <o> && date
<[[art]]> , <(beginnen|aufnehmen|abschließen|betreiben|absolvieren)> , <(Studium|.*studium)> \ <o=[[wissenschaft]]> == <art> has_studied <o> && date
<[[art]]> , <(erwerben|erlangen|erhalten|erreichen|machen)> , <(Abschluss|Abschluß|Bachelor|Master|Diplom|.*diplom|Magister|A. B.|M. A.|M. S.|B. A.|B. S.)> \ <o=[[wissenschaft]]> == <art> has_studied <o> && date
<[[art]]> , <(erwerben|erlangen|erhalten|erreichen|machen)> , <[[academic_degree]]> \ <o=[[wissenschaft]]> == <art> has_studied <o> && date
<[[art]]> , <studieren>, <o=[[wissenschaft]]> == <art> has_studied <o> && date
<[[art]]> , <(zuwenden|zuwandte)> , <Studium> \ <o=[[wissenschaft]]> == <art> has_studied <o> && date
<*> , <ausbilden> , <[[art]]> \ <in> <o=[[wissenschaft]]> == <art> has_studied <o> && date
<*> , <einschreiben> , <[[art]]> \ <für> <o=[[wissenschaft]]> == <art> has_studied <o> && date
<[[art]]> , <(beschreiben|schreiben)> , <in> <(Dissertation|Doktorarbeit)> \ <o=[[wissenschaft]]> == <art> has_subject_of_phd <o> && date
<[[art]]> , <(erwerben|erlangen|erhalten|erreichen|machen|sein)> , <(Doktor.*|Ph. D.)> \ <o=[[wissenschaft]]> == <art> has_subject_of_phd <o> && date
<[[art]]> , <promovieren> , <in> <o=[[wissenschaft]]> == <art> has_subject_of_phd <o> && date
<[[art]]> , <promovieren> , <zum Doktor> \ <o=[[wissenschaft]]> == <art> has_subject_of_phd <o> && date
<[[art]]> , <(sein|erhalten)> , <(Doktorand.*|Promovend.*)> \ <o=[[wissenschaft]]> == <art> has_subject_of_phd <o> && date
<[[art]]> , <(verfassen|schreiben|beginnen|abschließen)> , <(Dissertation|Doktorarbeit)> \ <o=[[wissenschaft]]> == <art> has_subject_of_phd <o> && date
<*> , <promovieren> , <[[art]]> \ <in> <o=[[wissenschaft]]> == <art> has_subject_of_phd <o> && date
<[[art]]> , <bekommen> , <o=[[award]]> == <art> has_won_award <o> && date
<[[art]]> , <erhalten> , <o=[[award]]> == <art> has_won_award <o> && date
```

<[[art]]> , <sein> , <(Träger.*|Preisträger.*)> \ <o=[[award]]> == <art> has_won_award <o> && date
<*> , <auszeichnen> , <[[art]]> \ <mit> <o=[[award]]> == <art> has_won_award <o> && date
* <o=[[award]]> , <(Ehrung.*|Auszeichnung.*|Preis.*)> == <art> has_won_award <o> && date
<*> , <verleihen> , <[[art]]> \ <o=[[award]]> == <art> has_won_award <o> && date
<[[art]]> , <sein> , <.*herausgeber.*> \ <o=[[journal]]> == <art> is_editor_of <o> && date
* <o=[[journal]]> , <.*herausgeber.*> == <art> is_editor_of <o> && date
<[[art]]> , <leiten> , <o=[[organization]]> == <art> is_executive_of <o> && date
<[[art]]> , <sein> , <(Präsident.*|Vorsitzende.*|Leiter.*|Direktor.*|Rektor.*|.*rektor.*|Kanzler.*|Dekan.*)> \ <o=[[organization]]> == <art> is_executive_of <o> && date
<[[art]]> , <übernehmen> , <(Leitung|Vorsitz)> \ <o=[[organization]]> == <art> is_executive_of <o> && date
<[[art]]> , <(vorstehen|präsidieren)> , <o=[[organization]]> == <art> is_executive_of <o> && date
<*> , <betrauen> , <[[art]]> \ <mit> <(Leitung|Vorsitz)> \ <o=[[organization]]> == <art> is_executive_of <o> && date
<*> , <(wählen|ernennen|berufen|aufnehmen)> , <[[art]]> \ <(Präsident.*|Vorsitzende.*|Leiter.*|Direktor.*|Rektor.*|.*rektor.*|Kanzler.*|Dekan.*)> \ <o=[[organization]]> == <art> is_executive_of <o> && date
<[[art]]> , <(sein|erhalten)> , <Ehrendoktor.*> \ <o=[[school]]> == <art> is_honorary_doctor_at <o> && date
* <Ehrendoktor.*> <o=[[school]]> , <(Auszeichnung.*|Ehrung.*|Preis.*)> == <art> is_honorary_doctor_at <o> && date
<o=[[school]]>, <verleihen> , <[[art]]> \ <Ehrendoktor.*> == <art> is_honorary_doctor_at <o> && date
<[[art]]> , <(erhalten|folgen|annehmen)> , <Ruf an> <o=[[school]]> == <art> is_lecturer_at <o> && date
<[[art]]> , <(haben|beziehen)> , <Lehrstuhl> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> is_lecturer_at <o> && date
<[[art]]> , <sein>, <(Dozent.*|.*dozent.*|Lizenziat.*|Lizentiat.*|Lecturer)> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> is_lecturer_at <o> && date
<[[art]]> , <(übernehmen|bekommen|erhalten|antreten)> , <(Dozentur.*|.*dozentur.*)> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> is_lecturer_at <o> && date
<[[art]]> , <(unterrichten|lehren)> , <(an|am|in|ans|ins)> <o=[[school]]> == <art> is_lecturer_at <o> && date
<[[art]]> , <(haben|beziehen)> , <Lehrstuhl> \ <in> <o=[[ort]]> == <art> is_lecturer_in <o> && date
<[[art]]> , <sein>, <(Dozent.*|.*dozent.*|Lizenziat.*|Lizentiat.*|Lecturer)> \ <in> <o=[[ort]]> == <art> is_lecturer_in <o> && date
<[[art]]> , <(übernehmen|bekommen|erhalten|antreten)> , <(Dozentur.*|.*dozentur.*)> \ <in> <o=[[ort]]> == <art> is_lecturer_in <o> && date
<[[art]]> , <(unterrichten|lehren)> , <in> <o=[[ort]]> == <art> is_lecturer_in <o> && date
<[[art]]> , <sein>, <(Dozent.*|.*dozent.*|Lizenziat.*|Lizentiat.*|Lecturer)> <für> <o=[[wissenschaft]]> == <art> is_lecturer_of <o> && date
<[[art]]> , <(übernehmen|bekommen|erhalten|antreten)> , <(Dozentur.*|.*dozentur.*)> <für> <o=[[wissenschaft]]> == <art> is_lecturer_of <o> && date
<[[art]]> , <(unterrichten|lehren)> , <o=[[wissenschaft]]> == <art> is_lecturer_of <o> && date
<[[art]]> , <angehören> , <o=[[association]]> == <art> is_member_of <o> && date
<[[art]]> , <beitreten> , <o=[[association]]> == <art> is_member_of <o> && date
<[[art]]> , <sein> , <(Mitglied|.*mitglied|Fellow)> \ <o=[[association]]> == <art> is_member_of <o> && date
<*> , <aufnehmen> , <[[art]]> \ <in> <o=[[association]]> == <art> is_member_of <o> && date
<*> , <aufnehmen> , <[[art]]> \ <in> <o=[[association]]> == <art> is_member_of <o> && date
* <Mitglied.*|.*mitglied|Fellow> <o=[[association]]> , <(Auszeichnung|Ehrung)> == <art> is_member_of <o> && date
* <o=[[association]]> , <Mitglied.*> == <art> is_member_of <o> && date
<*> , <(wählen|ernennen|berufen|aufnehmen)> , <[[art]]> \ <(als|zu|zum)> <Mitglied|.*mitglied|Fellow> \ <o=[[association]]> == <art> is_member_of <o> && date
<[[art]]> , <(arbeiten|entwickeln|entdecken|bauen|aufbauen|forschen|bleiben|zusammenarbeiten|mitarbeiten)>, <als> <(Professor.*|.*professor.*)> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> is_professor_at <o> && date
<[[art]]> , <sein>, <(Professor.*|.*professor.*)> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> is_professor_at <o> && date
<[[art]]> , <(übernehmen|bekommen|erhalten|antreten)> , <(Professur.*|.*professur.*)> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> is_professor_at <o> && date
<*> , <(ernennen|berufen)> , <[[art]]> \ <(zum|zu|als)> <(Professor.*|.*professor.*)> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> is_professor_at <o> && date
<[[art]]> , <(arbeiten|entwickeln|entdecken|bauen|aufbauen|forschen|bleiben|zusammenarbeiten|mitarbeiten)>, <als> <(Professor.*|.*professor.*)> \ <in> <o=[[ort]]> == <art> is_professor_in <o> && date
<[[art]]> , <sein>, <(Professor.*|.*professor.*)> \ <in> <o=[[ort]]> == <art> is_professor_in <o> && date
<[[art]]> , <(übernehmen|bekommen|erhalten|antreten)> , <(Professur.*|.*professur.*)> \ <in> <o=[[ort]]> == <art> is_professor_in <o> && date
<*> , <(ernennen|berufen)> , <[[art]]> \ <(zum|zu|als)> <(Professor.*|.*professor.*)> \ <in> <o=[[ort]]> == <art> is_professor_in <o> && date
<[[art]]> , <(arbeiten|entwickeln|entdecken|bauen|aufbauen|forschen|bleiben|zusammenarbeiten|mitarbeiten)>, <als> <(Professor.*|.*professor.*)> <für> <o=[[wissenschaft]]> == <art> is_professor_of <o> && date
<[[art]]> , <sein>, <(Professor.*|.*professor.*)> <für> <o=[[wissenschaft]]> == <art> is_professor_of <o> && date
<[[art]]> , <(übernehmen|bekommen|erhalten|antreten)> , <(Professur.*|.*professur.*)> <für> <o=[[wissenschaft]]> == <art> is_professor_of <o> && date
<*> , <(ernennen|berufen)> , <[[art]]> \ <(zum|zu|als)> <(Professor.*|.*professor.*)> <für> <o=[[wissenschaft]]> == <art> is_professor_of <o> && date
<[[art]]> , <lernen> , <(bei|unter|von)> <o=[[person]]> == <art> is_student_of <o> && date

<[[art]]> , <(machen|sein)> , <(Doktor.*|Ph. D.)> \ <(bei|unter)> <o=[[person]]> == <art> is_student_of <o> && date
<[[art]]> , <promovieren> , <als> <(Student.*|Schüler.*|Doktorand.*|Promovend.*)> <von> <o=[[person]]> == <art> is_student_of <o> && date
<[[art]]> , <promovieren> , <(bei|unter)> <o=[[person]]> == <art> is_student_of <o> && date
<[[art]]> , <sein> , <(Student.*|Schüler.*|Doktorand.*|Promovend.*)> \ <(bei|unter|von)> <o=[[person]]> == <art> is_student_of <o> && date
<[[art]]> , <studieren> , <(bei|unter)> <o=[[person]]> == <art> is_student_of <o> && date
<*> , <promovieren> , <[[art]]> \ <als> <(Student.*|Schüler.*|Doktorand.*|Promovend.*)> <von> <o=[[person]]> == <art> is_student_of <o> && date
<*> , <promovieren> , <[[art]]> \ <(bei|unter)> <o=[[person]]> == <art> is_student_of <o> && date
<[[art]]> , <(ziehen|übersiedeln|siedeln|gehen|kommen|wechseln|aufhalten|bleiben|wohnen|niederlassen|auswandern|zurückkehren|zurückgehen)> , <(in|auf|nach|aufs|ins)> <o=[[ort]]> == <art> lives_in <o> && date
<*> , <berufen> , <[[art]]> \ <(in|auf|nach|aufs|ins)> <o=[[ort]]> == <art> lives_in <o> && date
* <o=[[entities]]> , <siehe auch> == <art> see_also <o>
<*> , <adeln> , <[[art]]> == <art> was_ennobled_in <date>
<*> , <erheben> , <[[art]]> \ <in> <Adel.*> == <art> was_ennobled_in <date>
<*> , <schlagen> , <[[art]]> \ <(zum|zur|zu)> <[[peerage]]> == <art> was_ennobled_in <date>
<*> , <verhaften> , <[[art]]> == <art> was_imprisoned <date>
<*> , <verhaften> , <[[art]]> \ <in> <o=[[ort]]> == <art> was_imprisoned_in <o> && date
<[[art]]> , <(arbeiten|entwickeln|entdecken|bauen|aufbauen|forschen|zusammenarbeiten|mitarbeiten)> , <(an|am|in|ans|ins|bei|für)> <o=[[organization]]> == <art> works_at <o> && date
<[[art]]> , <beschäftigen> , <Forschung.*> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> works_at <o> && date
<[[art]]> , <(durchführen|beitreiben|machen)> , <(Versuch.*|.*versuch.*|Studie.*)> \ <(an|am|in|ans|ins|bei|für)> <o=[[organization]]> == <art> works_at <o> && date
<[[art]]> , <gehen> , <(an|am|in|ans|ins|zu|zum|zur)> <o=[[organization]]> == <art> works_at <o> && date
<[[art]]> , <sein> , <(anstellen|einstellen)> \ <(an|am|in|ans|ins|bei|für)> <o=[[organization]]> == <art> works_at <o> && date
<[[art]]> , <sein> , <(Assistent.*|Lehrer.*|Mitarbeiter.*|Kollege.*)> \ <(an|am|in|ans|ins|für|bei)> <o=[[organization]]> == <art> works_at <o> && date
<[[art]]> , <(sein|erhalten)> , <(postdoctoral|fellowship|Postdoc.*|Postdok.*)> \ <an> <o=[[organization]]> == <art> works_at <o> && date
<[[art]]> , <sein> , <(tätig|aktiv)> \ <(an|am|in|ans|ins|bei|für)> <o=[[organization]]> == <art> works_at <o> && date
<[[art]]> , <spezialisieren> , <(an|am|in|ans|ins|bei|für)> <o=[[organization]]> == <art> works_at <o> && date
<[[art]]> , <(übernehmen|annehmen|erhalten)> , <(Stelle.*|.*stelle.*|Stellung|Anstellung|Posten|Position|Leitung|Vorsitz)> \ <(an|am|in|ans|ins|bei|für)> <o=[[organization]]> == <art> works_at <o> && date
<[[art]]> , <(wechseln|niederlassen|siedeln)> , <(an|am|in|ans|ins)> <o=[[organization]]> == <art> works_at <o> && date
<[[art]]> , <wirken> , <(an|am|in|ans|ins|bei|für)> <o=[[organization]]> == <art> works_at <o> && date
<*> , <berufen> , <[[art]]> \ <(an|am|in|ans|ins)> <o=[[school]]> == <art> works_at <o> && date
<*> , <betrauen> , <[[art]]> \ <mit> <(Stelle.*|.*stelle.*|Stellung|Anstellung|Posten|Position|Leitung|Vorsitz)> \ <(an|am|in|ans|ins|zu|zum|zur)> <o=[[organization]]> == <art> works_at <o> && date
<*> , <holen> , <[[art]]> \ <(an|am|in|ans|ins|bei|für)> <o=[[organization]]> == <art> works_at <o> && date
<[[art]]> , <(arbeiten|betreiben|forschen|gründen)> , <in> <o=[[ort]]> == <art> works_in <o> && date
<[[art]]> , <(durchführen|beitreiben|machen)> , <(Versuch.*|.*versuch.*|Studie.*)> \ <in> <o=[[ort]]> == <art> works_in <o> && date
<[[art]]> , <erhalten> , <Anstellung> \ <in> <o=[[ort]]> == <art> works_in <o> && date
<[[art]]> , <sein> , <(Assistent.*|Lehrer.*|Mitarbeiter.*|Kollege.*)> \ <in> <o=[[ort]]> == <art> works_in <o> && date
<[[art]]> , <sein> , <(Präsident.*|Vorsitzende.*|Leiter.*|Direktor.*|Rektor.*|.*rektor.*|Kanzler.*|Dekan.*)> \ <in> <o=[[ort]]> == <art> works_in <o> && date
<[[art]]> , <sein> , <(tätig|aktiv)> \ <in> <o=[[ort]]> == <art> works_in <o> && date
<[[art]]> , <(übernehmen|annehmen|erhalten)> , <(Stelle.*|.*stelle.*|Stellung|Anstellung|Posten|Position|Leitung|Vorsitz)> \ <in> <o=[[ort]]> == <art> works_in <o> && date
<[[art]]> , <wirken> , <in> <o=[[ort]]> == <art> works_in <o> && date
<*> , <betrauen> , <[[art]]> \ <mit> <(Stelle.*|.*stelle.*|Stellung|Anstellung|Posten|Position|Leitung|Vorsitz)> \ <in> <o=[[ort]]> == <art> works_in <o> && date
<*> , <(wählen|ernennen|berufen|aufnehmen)> , <[[art]]> \ <(Präsident.*|Vorsitzende.*|Leiter.*|Direktor.*|Rektor.*|.*rektor.*|Kanzler.*|Dekan.*)> \ <in> <o=[[ort]]> == <art> works_in <o> && date
<[[art]]> , <(arbeiten|entwickeln|entdecken|bauen|aufbauen|forschen|zusammenarbeiten|mitarbeiten|untersuchen|konstruieren)> , <als> <Assistent.*> <o=[[person]]> == <art> works_with <o> && date
<[[art]]> , <(arbeiten|entwickeln|entdecken|bauen|aufbauen|forschen|zusammenarbeiten|mitarbeiten|untersuchen|konstruieren)> , <als> <Assistent.*> <von> <o=[[person]]> == <art> works_with <o> && date
<[[art]]> , <(arbeiten|entwickeln|entdecken|bauen|aufbauen|forschen|zusammenarbeiten|mitarbeiten|untersuchen|konstruieren)> , <(bei|mit|unter)> <o=[[person]]> == <art> works_with <o> && date
<[[art]]> , <(assistieren|zuarbeiten)> , <o=[[person]]> == <art> works_with <o> && date
<[[art]]> \ <o=[[person]]> , <(verfassen|schreiben)> , <(zusammen|gemeinsam)> == <art> works_with <o> && date
<[[art]]> , <sein> , <anstellen> \ <(bei|für|unter|von)> <o=[[person]]> == <art> works_with <o> && date
<[[art]]> , <sein> , <(Assistent.*|Mitarbeiter.*|Kollege.*)> <o=[[person]]> == <art> works_with <o> && date
<[[art]]> , <sein> , <(Assistent.*|Mitarbeiter.*|Kollege.*)> <von> <o=[[person]]> == <art> works_with <o> && date

```
<[[art]]> , <sein> , <(tätig|aktiv)> \ <(bei|für|unter)> <o=[[person]]> == <art> works_with <o> && date
<[[art]]> , <(verfassen|schreiben)> , <mit> <o=[[person]]> == <art> works_with <o> && date
<[[art]]> , <(verfassen|schreiben)> , <(zusammen|gemeinsam)> \ <o=[[person]]> == <art> works_with <o> && date
* <mit> <o=[[person]]> , <Schrift> == <art> works_with <o> && date
<[[art]]> , <sein> , <Lehrer.*> <o=[[person]]> == <o> is_student_of <art> && date
<[[art]]> , <sein> , <Lehrer.*> <von> <o=[[person]]> == <o> is_student_of <art> && date
<*> , <(benennen|umbenennen)> , <o=[[entities]]> \ <nach> <[[art]]> == <o> named_after <art> && date
<*> , <(benennen|umbenennen)> , <o=[[entities]]> \ <zu Ehre> \ <[[art]]> == <o> named_after <art> && date
* <o=[[entities]]> , <Namensgeber> == <o> named_after <art> && date
```

B Klassifizierungsregeln `classifiers-wissenschaftler.txt`

```
Category:/^Ort\b/ -> ort
Category:/^Stadt\b/ -> ort
Category:/^Gemeinde\b/ -> ort
Category:/^Staat\b/ -> ort
Category:/^Region\b/ -> ort
Category:/^Land\b/ -> ort
Category:/^Schule\b/ -> school
Category:/^Hochschule\b/ -> school
Category:/^Universität\b/ -> school
Category:/^Fachhochschule\b/ -> school
Category:/wissenschaft\b/ -> wissenschaft
Category:/^Wissenschaft\b/ -> wissenschaft
Category:/^Wissenschaftliches Fachgebiet\b/ -> wissenschaft
Category:/Studienfach\b/ -> wissenschaft
Category:/Unternehmen\b/ -> company
Category:/Firma\b/ -> company
Category:/^Mann\b/ -> male_person
Category:/^Frau\b/ -> female_person
Category:/preis\b/ -> award
Category:/Preis\b/ -> award
Category:/Ehrenzeichen\b/ -> award
Category:/Orden\b/ -> award
Category:/Auszeichnung\b/ -> award
Category:/[Vv]erein\b/ -> association
Category:/[Vv]erband\b/ -> association
Category:/Organisation\b/ -> association
Category:/Stiftung\b/ -> association
Category:/[Vv]ereinigung\b/ -> association
Category:/[CK]lub\b/ -> association
Category:/[ck]lub\b/ -> association
Category:/[Mm]useum\b/ -> museum
Category:/Observatorium\b/ -> observatory
Category:/Adelstitel\b/ -> peerage
Category:/[Zz]eitschrift/ -> journal
Category:/[Zz]eitung/ -> journal
Category:/Akademischer Grad\b/ -> academic_degree
[[male_person]] -> person
[[female_person]] -> person
[[company]] -> organization
[[museum]] -> organization
[[school]] -> organization
[[association]] -> organization
[[observatory]] -> organization
```

C Extrahierte Tripel pro Artikel in der Kategorie «Wissenschaftler»

Anzahl Tripel	Zahl der Artikel (absolut)	Zahl der Artikel (in %)	**32**	3	0.004
0	12133	17.631	**33**	6	0.009
1	10113	14.695	**34**	6	0.009
2	9453	13.736	**35**	7	0.01
3	8400	12.206	**36**	5	0.007
4	6682	9.71	**37**	1	0.001
5	5449	7.918	**38**	1	0.001
6	4099	5.956	**39**	0	0
7	3062	4.449	**40**	1	0.001
8	2387	3.469	**41**	3	0.004
9	1695	2.463	**42**	0	0
10	1353	1.966	**43**	3	0.004
11	988	1.436	**44**	1	0.001
12	716	1.04	**45**	0	0
13	568	0.825	**46**	1	0.001
14	383	0.557	**47**	2	0.003
15	307	0.446	**48**	0	0
16	232	0.337	**49**	2	0.003
17	157	0.228	**50**	0	0
18	113	0.164	**51**	2	0.003
19	99	0.144	**52**	0	0
20	87	0.126	**53**	1	0.001
21	70	0.102	**54**	0	0
22	54	0.078	**55**	0	0
23	32	0.046	**56**	0	0
24	38	0.055	**57**	0	0
25	22	0.032	**58**	0	0
26	16	0.023	**59**	1	0.001
27	13	0.019	**60**	3	0.004
28	14	0.02	**61**	0	0
29	13	0.019	**62**	2	0.003
30	11	0.016	**63**	0	0
31	7	0.01	**64**	1	0.001

D 200 Tripel der Zufallsstichprobe (in Präfixschreibweise)

Präfixe

```
@prefix prop: <http://wiki2rdf.ibi.hu-berlin.de/property/> .
@prefix db: <http://de.dbpedia.org/resource/> .
@prefix datetime: <http://wiki2rdf.ibi.hu-berlin.de/datetime/> .
```

Subjekt	Prädikat	Objekt	Zeit (*context*)
db:Friedrich_Nölke	prop:has_studied	db:Physik	datetime:1899/1902
db:Hugo_von_Mohl	prop:works_at	db:Universität_München	datetime:1828
db:Margret_Nemann	prop:has_studied_at	db:Westfälische_Wilhelms-Universität	datetime:1974
db:Michael_Keller_(Bischof)	prop:has_studied_in	db:Rom	datetime:null
db:Christoph_Safferling	prop:works_at	db:Universität_Hannover	datetime:2000/2002
db:Pierre_Tercier	prop:is_honorary_doctor_at	db:Universität_Paris_II	datetime:2004
db:Heinz_Kußmaul	prop:is_professor_at	db:Universität_des_Saarlandes	datetime:null
db:Karl_Heinrich_Lange	prop:has_studied_at	db:Universität_Jena	datetime:1720
db:Alan_Turing	prop:lives_in	db:London	datetime:1912-06-23
db:William_Trager	prop:is_professor_at	db:Harvard_University	datetime:1980
db:Ernst_Kloss	prop:is_student_of	db:Heinrich_Wölfflin	datetime:1923
db:Paul_Karrer	prop:works_in	db:Zürich	datetime:null
db:Percy_Heawood	prop:works_at	db:Durham_University	datetime:1887
db:Heinrich_Kreutz_(Astronom)	prop:works_with	db:Theodor_Oppolzer	datetime:null
db:Trevor_Wooley	prop:has_won_award	db:Berwick-Preis	datetime:1993
db:Tadija_Smičiklas	prop:is_lecturer_in	db:Rijeka	datetime:null
db:Konzilstheologe	prop:see_also	db:Augustin_Bea	datetime:null
db:Helge_Klaus_Rieder	prop:has_place_of_phd	db:Universität_Bamberg	datetime:1992
db:Paul_Demiéville_(Sinologe)	prop:is_professor_at	db:Collège_de_France	datetime:1946/1964
db:Edward_A._Thompson	prop:lives_in	db:Swansea	datetime:1941
db:Emil_Lampe	prop:is_student_of	db:Ernst_Eduard_Kummer	datetime:1864
db:Albert_von_Bamberg	prop:is_member_of	db:Akademie_gemeinnütziger_Wissenschaften	datetime:1899
db:Erwin_Panofsky	prop:is_student_of	db:Wilhelm_Vöge	datetime:1914
db:Henri_Ellenberger	prop:works_in	db:Topeka_(Kansas)	datetime:null
db:Franz_Boas	prop:is_honorary_doctor_at	db:Universität_Kiel	datetime:1931

db:Philip_J._Currie	prop:works_in	db:Argentinien	datetime:null
db:Peter_Spahn_(Historiker)	prop:is_lecturer_of	db:Sozialgeschichte	datetime:1988
db:Alfred_Lotze_(Mathematiker)	prop:has_studied_at	db:Universität_Tübingen	datetime:1901
db:Johann_Carl_Gehler	prop:has_place_of_phd	db:Leipzig	datetime:1758
db:Ingrid_Mielenz	prop:has_studied_at	db:Freie_Universität_Berlin	datetime:1966
db:Heinrich_Ewers	prop:has_studied	db:Rechtswissenschaften	datetime:1948
db:Richard_Willstätter	prop:works_at	db:Ludwig-Maximilians-Universität_München	datetime:null
db:Franz_Sales_Sklenitzka	prop:has_won_award	db:Zehn_besondere_Bücher_zum_Andersentag	datetime:2006
db:Andreas_Gold	prop:has_studied_at	db:Universität_Heidelberg	datetime:1976/1982
db:Adolf_Zade	prop:lives_in	db:Stockholm	datetime:null
db:Laurentius_Kirchhoff	prop:has_place_of_phd	db:Universität_Köln	datetime:null
db:Walther_Hinz	prop:has_place_of_phd	db:Leipzig	datetime:1930
db:Émile_Chevé	prop:lives_in	db:Paris	datetime:1835
db:Charles_Patin	prop:lives_in	db:Padua	datetime:null
db:Hugo_Hegeland	prop:has_studied_at	db:Universität_Göteborg	datetime:1942
db:Joachim_Sterck_van_Ringelbergh	prop:lives_in	db:Orléans	datetime:????-01
db:Johannes_Gründel	prop:is_member_of	db:Cartellverband_der_katholischen_deutschen_Studentenverbindungen	datetime:null
db:Jeremy_Kahn	prop:has_studied_at	db:Harvard_University	datetime:1995
db:Karl_Schlabow	prop:works_with	db:Herbert_Jankuhn	datetime:null
db:Viktor_Pöschl	prop:is_lecturer_in	db:Ruprecht-Karls-Universität_Heidelberg	datetime:1940
db:Ernst_Bloch	prop:has_won_award	db:Friedenspreis_des_Deutschen_Buchhandels	datetime:1970
db:Franz_Kern	prop:lives_in	db:Stettin	datetime:1852
db:Heinrich_Philipp_Konrad_Henke	prop:has_studied_in	db:Braunschweig	datetime:1772
db:Hermann_Ignatius_Flender	prop:works_in	db:Aschaffenburg	datetime:1736-02
db:Bernhard_Hassenstein	prop:has_won_award	db:Max_Born-Medaille_für_Verantwortung_in_der_Wissenschaft	datetime:1981
db:Wolfgang_Darsow	prop:has_studied_at	db:Universität_zu_Köln	datetime:null
db:Serafím_Fernandes_de_Araújo	prop:has_studied_in	db:Rom	datetime:null
db:Claus_Leitzmann	prop:is_member_of	db:Deutsche_Gesellschaft_für_Ernährung	datetime:null
db:Pei_Wenzhong	prop:is_member_of	db:Chinesische_Akademie_der_Wissenschaften	datetime:null
db:Balthasar_Hubmaier	prop:is_student_of	db:Johannes_Eck	datetime:null
db:William_Montgomery_Watt	prop:is_member_of	db:Iona_Community	datetime:null
db:Alessandro_Barchiesi	prop:is_professor_at	db:Universität_Siena	datetime:2000
db:Rainer_Moritz	prop:has_studied	db:Germanistik	datetime:null
db:Jürgen_Petersohn	prop:is_executive_of	db:Konstanzer_Arbeitskreis_für_mittelalterliche_Geschichte	datetime:1998/2001
db:Horst_Bourmer	prop:is_executive_of	db:Marburger_Bund	datetime:1961/1968
db:Gottfried_Köthe	prop:is_honorary_doctor_at	db:Westfälische_Wilhelms-Universität	datetime:null

db:Dhori_Kule	prop:works_in	db:Kuçova	datetime:null
db:Song_Du-yul	prop:is_professor_in	db:Heidelberg	datetime:1982
db:Christian_Wandrey	prop:is_executive_of	db:Forschungszentrum_Jülich	datetime:1979/2008
db:Hasso_Plattner	prop:has_founded	db:SAP_AG	datetime:1972
db:Heinrich_Drerup	prop:is_lecturer_at	db:Westfälische_Wilhelms-Universität	datetime:null
db:Matthias_Varga_von_Kibéd	prop:works_at	db:Universität_München	datetime:null
db:Wolfgang_Giloi	prop:is_honorary_doctor_at	db:Technische_Universität_Darmstadt	datetime:2003
db:Axel_Buchner	prop:is_executive_of	db:Heinrich-Heine-Universität_Düsseldorf	datetime:null
db:Eli_Fischer-Jørgensen	prop:has_studied_at	db:Universität_Kopenhagen	datetime:1933/1936
db:Dan_Olweus	prop:is_professor_at	db:Universität_Bergen	datetime:null
db:Conrad_Müller	prop:works_with	db:Georg_Prange	datetime:null
db:Roland_Faber	prop:is_member_of	db:Internationale_Gesellschaft_für_Neue_Musik	datetime:1989
db:Hans-Dieter_Evers	prop:has_studied	db:Wirtschaftswissenschaften	datetime:null
db:Rainer_Schnell	prop:is_professor_at	db:Universität_Duisburg-Essen	datetime:2008
db:Matthäus_Adriani	prop:lives_in	db:Basel	datetime:1478/1556
db:Karl_A._E._Enenkel	prop:has_studied_at	db:Universität_von_Amsterdam	datetime:1977/1979
db:Heinrich_Hunke	prop:works_with	db:Leonardo_Conti_(Mediziner)	datetime:1939
db:Stephen_Pisano	prop:works_at	db:Päpstliches_Bibelinstitut	datetime:1982
db:Ägidius_Hunnius_der_Jüngere	prop:has_place_of_phd	db:Sangerhausen	datetime:1623
db:Willem_Levelt	prop:has_won_award	db:Pour_le_Mérite	datetime:2010
db:Haworth-Formel	prop:named_after	db:Walter_Norman_Haworth	datetime:null
db:Golo_Mann	prop:has_won_award	db:Kultureller_Ehrenpreis_der_Landeshauptstadt_München	datetime:1980
db:Ferdinand_von_Roemer	prop:is_member_of	db:Royal_Society	datetime:1859
db:Andreas_Schreitmüller	prop:is_professor_at	db:Universität_Konstanz	datetime:2000-10
db:Kai_Brodersen	prop:works_with	db:Jas_Elsner	datetime:2009
db:Hellmut_Becker	prop:is_member_of	db:NSDAP	datetime:????-05
db:Azaria_dei_Rossi	prop:lives_in	db:Sabbioneta	datetime:null
db:Heinrich_Friedjung	prop:is_honorary_doctor_at	db:Universität_Heidelberg	datetime:null
db:Ludwig_von_Strümpell	prop:is_student_of	db:Johann_Friedrich_Herbart	datetime:1833
db:Paul_Nurse	prop:works_at	db:University_of_Sussex	datetime:null
db:Theo_Geisel	prop:has_won_award	db:Gottfried-Wilhelm-Leibniz-Preis	datetime:1994
db:Hinrich_Seidel	prop:has_won_award	db:Ehrenlegion	datetime:1992
db:Alexander_Petrowitsch_Karpinski	prop:is_member_of	db:Russische_Akademie_der_Wissenschaften	datetime:1886
db:Joseph_Nye	prop:is_executive_of	db:Trilaterale_Kommission	datetime:null
db:Oswald_Huber	prop:is_professor_at	db:Universität_Freiburg_(Schweiz)	datetime:1989
db:Friedrich_Schrader	prop:works_at	db:Robert_College	datetime:1891

db:Jean-Claude_Hollerich	prop:has_place_of_phd	db:Bonn	datetime:2001
db:Jan_Schröder_(Rechtswissenschaftler)	prop:is_editor_of	db:Zeitschrift_für_Neuere_Rechtsgeschichte	datetime:null
db:Klaus_Hornung	prop:has_won_award	db:Bundesverdienstkreuz	datetime:null
db:Paul_Radin	prop:lives_in	db:Lugano	datetime:1952
db:Gunnar_Carlsson	prop:has_studied_at	db:Harvard_University	datetime:1973
db:Dino_del_Garbo	prop:lives_in	db:Florenz	datetime:1319
db:Robert_Schwarz	prop:works_at	db:Technische_Hochschule_Berlin	datetime:null
db:Theodor_W._Adorno	prop:see_also	db:Verblendungszusammenhang	datetime:null
db:John_A._Rogers	prop:is_member_of	db:American_Physical_Society	datetime:null
db:Ivan_Vidav	prop:has_won_award	db:Prešeren-Preis	datetime:1952
db:Maixent_Coly	prop:has_studied_at	db:Päpstliches_Bibelinstitut	datetime:1983/1987
db:Christoph_Gröpl	prop:is_professor_at	db:Deutsche_Hochschule_für_Verwaltungswissenschaften_Speyer	datetime:2003
db:Rolf_Rendtorff	prop:works_in	db:Göttingen	datetime:1962
db:Johann_Paul_Kress	prop:has_studied	db:Rechtswissenschaft	datetime:null
db:Markus_Schauer	prop:works_at	db:Freie_Universität_Berlin	datetime:2006
db:Jean-Paul_Vinay	prop:is_lecturer_at	db:University_of_Victoria	datetime:1966/1976
db:Carl_von_Linde	prop:see_also	db:Kühlschrank	datetime:null
db:Johann_Heinrich_Schulze	prop:has_studied	db:Theologie	datetime:1704/1717
db:Jan_Bašta	prop:has_studied_in	db:Jičín	datetime:1878/1884
db:Waldemar_Ilberg	prop:has_studied_at	db:Königin-Carola-Gymnasium	datetime:1916
db:Christian_Heinrich_Hecht	prop:lives_in	db:Sosa_(Eibenstock)	datetime:1772
db:John_Lawson_(Physiker)	prop:works_in	db:Harwell	datetime:1947
db:Pierre_Samuel_du_Pont_de_Nemours	prop:was_imprisoned	datetime:date	datetime:date
db:Walter_Paska	prop:has_studied	db:Philosophie	datetime:null
db:Frank_J._Dixon	prop:has_studied_at	db:University_of_Minnesota	datetime:1943
db:Jack_R._Pole	prop:works_at	db:Princeton_University	datetime:1952/1953
db:Defterdar_Sarı_Mehmet_Pascha	prop:see_also	db:Liste_Osmanischer_Chronisten	datetime:null
db:Jan_Waldenström	prop:is_professor_at	db:Universität_Uppsala	datetime:1947
db:Joseph_Charles_Arthur	prop:has_studied_at	db:Johns_Hopkins_University	datetime:null
db:Carl_O._Dunbar	prop:works_with	db:Charles_Schuchert	datetime:1949
db:Peter_Nijkamp	prop:has_won_award	db:Spinoza-Preis	datetime:1996
db:Hermann_Denz	prop:has_studied_in	db:Bregenz	datetime:1967/1971
db:Anton_von_Henle	prop:is_member_of	db:Kartellverband_katholischer_deutscher_Studentenvereine	datetime:null
db:Edward_James_McShane	prop:works_with	db:John_L._Kelley	datetime:1953
db:Wolfgang_Heinz_(Kriminologe)	prop:is_professor_at	db:Universität_Konstanz	datetime:1981
db:Werner_Buttler	prop:is_member_of	db:Schutzstaffel	datetime:1935

db:Christian_Hesse	prop:is_professor_at	db:Universität_Stuttgart	datetime:1991
db:Georg_Liebscher	prop:works_in	db:Magdeburg	datetime:null
db:Karl_Heinrich_Rau	prop:is_lecturer_at	db:Ruprecht-Karls-Universität_Heidelberg	datetime:1822
db:Karl_Friedrich_Meyer_(Pathologe)	prop:is_professor_at	db:University_of_Pennsylvania	datetime:null
db:James_R._Wilson	prop:works_with	db:Hans_Bethe	datetime:null
db:Josef_Kraus_(Lehrer)	prop:has_studied_in	db:Würzburg	datetime:null
db:Ynes_Mexia	prop:has_studied_at	db:University_of_California,_Berkeley	datetime:null
db:Eugen_Kalkschmidt	prop:works_in	db:Dresden	datetime:null
db:Michael_Freund	prop:lives_in	db:Kiel	datetime:1950
db:Emil_Jacobs_(Bibliothekar)	prop:works_at	db:Universität_Greifswald	datetime:1892
db:Hermann_Heller_(Jurist)	prop:is_professor_at	db:Johann_Wolfgang_Goethe-Universität	datetime:1932
db:William_Holland_(Asienforscher)	prop:lives_in	db:Amherst_(Massachusetts)	datetime:1990
db:Bartholomäus_Stein	prop:is_student_of	db:Konrad_Celtis	datetime:null
db:William_H._Miller	prop:is_member_of	db:National_Academy_of_Sciences	datetime:1987
db:Atto_Vannucci	prop:is_lecturer_of	db:Universalgeschichte	datetime:1852
db:Christian_Thiel	prop:is_member_of	db:Leopoldina	datetime:1993
db:Liste_bekannter_Neogräzisten	prop:see_also	db:Liste_griechischer_Schriftsteller_(Neuzeit)	datetime:null
db:Ludwig_Eichinger	prop:has_place_of_phd	db:Universität_Bayreuth	datetime:1980
db:Volker_Strassen	prop:has_won_award	db:Knuth-Preis	datetime:2008
db:Erich_Woytek	prop:has_place_of_phd	db:Wien	datetime:1969
db:Willi_A._Boelcke	prop:has_studied	db:Germanistik	datetime:1949
db:Ludwig_F._Haber	prop:has_studied_at	db:St_Paul's_School_(London)	datetime:null
db:Wladimir_Iwanowitsch_Wernadski	prop:is_professor_at	db:Lomonossow-Universität	datetime:1898/1911
db:Pawel_Samuilowitsch_Urysohn	prop:see_also	db:Urysohn-Raum	datetime:null
db:Kenneth_Morgan_Abbott	prop:works_at	db:Ohio_State_University	datetime:null
db:Max_Reich	prop:lives_in	db:Göttingen	datetime:null
db:Gary_Stormo	prop:has_place_of_phd	db:University_of_Colorado_at_Boulder	datetime:null
db:Erwin_Schrödinger	prop:works_with	db:Victor_Weisskopf	datetime:null
db:Gerhard_Ertl	prop:has_won_award	db:Verdienstorden_des_Landes_Baden-Württemberg	datetime:????-04
db:Thomas_Hales	prop:is_professor_at	db:University_of_Pittsburgh	datetime:2001
db:Valentin_Forster	prop:is_professor_at	db:Universität_Marburg	datetime:1569
db:Richard_Egenter	prop:is_professor_in	db:Passau	datetime:1933
db:Johann_Nepomuk_Hubert_von_Schwerz	prop:has_founded	db:Universität_Hohenheim	datetime:1818
db:Ludwig_Salgo	prop:has_studied_at	db:Eberhard_Karls_Universität_Tübingen	datetime:1968
db:Klaus_M._Leisinger	prop:is_executive_of	db:Stiftungsrat	datetime:2003
db:Arnold_Berney	prop:lives_in	db:Breslau	datetime:1933

db:Herbert_Busemann	prop:has_won_award	db:Lobatschewski-Medaille	datetime:1984
db:Lothar_Dresen	prop:is_member_of	db:Deutsche_Geophysikalische_Gesellschaft	datetime:null
db:Kurt_Blome	prop:works_at	db:Justus-Liebig-Universität_Gießen	datetime:1912
db:Caspar_Ratzenberger	prop:has_studied_in	db:Wittenberg	datetime:null
db:John_Williamson_(Wirtschaftswissenschaftler)	prop:is_lecturer_at	db:Universität_York	datetime:1963/1968
db:Bernd_Schönemann	prop:is_professor_at	db:Westfälische_Wilhelms-Universität	datetime:2004
db:Veit_Schiffmann	prop:has_studied_at	db:Universität_für_angewandte_Kunst_Wien	datetime:1979/1980
db:Albert_Einstein	prop:is_member_of	db:Accademia_Nazionale_dei_Lincei	datetime:1945
db:Michael_Schefczyk	prop:has_place_of_phd	db:RWTH_Aachen	datetime:1994-05
db:Res_Jost	prop:works_with	db:Wolfgang_Pauli	datetime:null
db:Bernhard_Felmberg	prop:is_member_of	db:Evangelischer_Entwicklungsdienst	datetime:null
db:Joachim_Latacz	prop:works_at	db:Thesaurus_Linguae_Graecae_(Hamburg)	datetime:1960/1966
db:Erin_Manning	prop:has_place_of_phd	db:Universität_von_Hawaii	datetime:2000
db:Walter_Kolneder	prop:has_won_award	db:Kulturpreis_des_Landes_Oberösterreich	datetime:1986
db:Stemonitaceae	prop:named_after	db:Oscar_Brefeld	datetime:null
db:Axel_von_Fersen	prop:is_student_of	db:Sven_Lagerbring	datetime:null
db:Eduard_Fein	prop:works_in	db:Braunschweig	datetime:1838
db:Fried_Mühlberg	prop:has_place_of_phd	db:Köln	datetime:1949
db:Friedrich_Vogt_(Germanist)	prop:is_professor_at	db:Universität_Greifswald	datetime:1874
db:Volker_Heine	prop:works_with	db:Marvin_Cohen	datetime:????-06
db:Karl_Rohe	prop:is_executive_of	db:Deutsche_Gesellschaft_für_Politikwissenschaft	datetime:1995
db:Georg_Braulik	prop:is_member_of	db:Österreichische_Akademie_der_Wissenschaften	datetime:1999
db:Theodor_Wintges	prop:is_lecturer_of	db:Kartografie	datetime:1987-03
db:Richard_Erdoes	prop:has_studied	db:Archäologie	datetime:null
db:Irving_Kaplansky	prop:works_at	db:University_of_Chicago	datetime:1945
db:Hans_Föllmer	prop:is_professor_at	db:Johann_Wolfgang_Goethe-Universität_Frankfurt_am_Main	datetime:1969/1970
db:Christian_Berger_(Jurist)	prop:is_professor_at	db:Universität_Leipzig	datetime:null
db:Albert_Lester_Lehninger	prop:is_professor_in	db:Baltimore	datetime:1952/1978
db:Walter_Layton,_1._Baron_Layton	prop:has_studied_at	db:Westminster_School	datetime:null
db:Lars_Gårding	prop:works_with	db:Arthur_Strong_Wightman	datetime:null
db:Ray_Tomlinson	prop:works_in	db:Cambridge_(Massachusetts)	datetime:1968

E Ergebnisse der SPARQL-Abfragen (ohne UTF8-Zeichen)

Personen, die in Köln studiert haben, sortiert nach Datum

?s	?g
db:Heymericus_de_Campo	datetime:1423
db:Heinrich_von_Rubenach	datetime:1450
db:Hieronymus_Dungersheim	datetime:1496
db:Kilian_Reuter	datetime:1507
db:Petrus_Mosellanus	datetime:1509/1511
db:Berchtold_Haller	datetime:1510
db:Johann_Dietenberger	datetime:1511
db:Johannes_Pollius	datetime:1516
db:Johann_Stammel	datetime:1535
db:Gerhard_Kleinsorgen	datetime:1548
db:Justus_Lipsius	datetime:1559
db:Bartholomaus_des_Bosses	datetime:1686
db:Johannes_Jakobus_Bouget	datetime:1782/1784
db:Johann_Heinrich_Achterfeld	datetime:1813
db:Karl_Hoffmeister	datetime:1816
db:Karl_Hoffmeister	datetime:1821
db:Karl_Hoffmeister	datetime:1832
db:Karl_Hoffmeister	datetime:1834
db:Karl_Hoffmeister	datetime:1842
db:Hermann_Kortum	datetime:1855
db:Wilhelm_Lexis	datetime:1855
db:Hermann_Joseph_Schmitz	datetime:1860
db:Joseph_Rebbert	datetime:1871
db:Berthold_Laufer	datetime:1884/1893
db:Adolf_Horion	datetime:1907/1910
db:Theodor_Steinbuchel	datetime:1908/1912
db:Paul_Hankamer	datetime:1910/1914
db:Erwin_Geldmacher	datetime:1912
db:Ernst_Alfred_Philippson	datetime:1918
db:Edmund_Beyenburg	datetime:1919
db:Heinrich_Bernds	datetime:1921
db:Wilhelm_Hasenack	datetime:1923
db:Benno_von_Wiese	datetime:1923/1924
db:Erwin_Metzke	datetime:1925/1929
db:Wilfrid_Schreiber	datetime:1927
db:Klaus_Morsdorf	datetime:1928/1931
db:Josef_Naas	datetime:1928/1933
db:Hans_Ebeling_(Publizist)	datetime:1929
db:Otto_Wilhelm_von_Vacano	datetime:1929
db:Alfred_Kamphausen	datetime:1929

db:Heinrich_Freiherr_von_Stackelberg	datetime:1930
db:Richard_Fritz_Behrendt	datetime:1931
db:Ernst_Hoffmann_(Philosoph)	datetime:1932
db:Joseph_Walk	datetime:1932/1933
db:Heinrich_Freiherr_von_Stackelberg	datetime:1934
db:Horst_Bourmer	datetime:1938
db:Alfons_M._Dauer	datetime:1941
db:Karl_Josef_Becker	datetime:1946/1948
db:Ernst_Gunther_Grimme	datetime:1947/1954
db:Karl_Josef_Becker	datetime:1948-04-13
db:Kurt_Kluxen	datetime:1949
db:Karl-Heinrich_Hansmeyer	datetime:1950/1954
db:Jochen_Hild	datetime:1951
db:Wolfgang_Kluxen	datetime:1951
db:Gunter_Birtsch	datetime:1951
db:Georg_Rudolf_Lind	datetime:1952
db:Werner_Pfeiffer	datetime:1953/1955
db:Wolfgang_Schieder	datetime:1954
db:Gunter_Wiegelmann	datetime:1954
db:Gunter_Herrmann_(Intendant)	datetime:1954/1957
db:Wolfgang_Speyer	datetime:1954/1958
db:Peter_Schonhofer	datetime:1955
db:Kurt_Abels	datetime:1955/1969
db:Ulrich_Karthaus	datetime:1956
db:Theodor_Berchem	datetime:1956
db:Ewald_Schmeken	datetime:1956
db:Joachim_Vobbe	datetime:1957/1966
db:Fred_Weidmann	datetime:1957/1970
db:Wolfgang_J._Mommsen	datetime:1958
db:Rolf_Bergmann	datetime:1958/1963
db:Jochen_Greven	datetime:1959
db:Gerhard_Scherhorn	datetime:1959
db:Diether_Roderich_Reinsch	datetime:1959
db:Hans_Medick	datetime:1959/1966
db:Stephan_Schroer	datetime:1960
db:Helmut_R._Leppien	datetime:1960
db:Dieter_Hagedorn	datetime:1961
db:Albrecht_Encke	datetime:1961
db:Maximilian_Wallerath	datetime:1961/1965
db:Alfred_Czarnetzki	datetime:1961/1966
db:Manfred_Paul_Dierich	datetime:1961/1966
db:Wolfgang_Bergsdorf	datetime:1961/1970
db:Albert_Henrichs	datetime:1962
db:Diether_Roderich_Reinsch	datetime:1962
db:Reinhold_Kaiser	datetime:1962/1968
db:Winfried_Schlepphorst	datetime:1963

db:Erling_von_Mende	datetime:1963
db:Gisela_Hellenkemper_Salies	datetime:1963
db:Rainer_K._Wick	datetime:1963/1969
db:Stephan_Schroer	datetime:1964
db:Peter_Gaehtgens	datetime:1964
db:Joachim_Starbatty	datetime:1964
db:Dieter_Liewerscheidt	datetime:1964/1969
db:Dirk_Hoeges	datetime:1964/1972
db:Jurgen_Koebke	datetime:1965
db:Claus_Offe	datetime:1965
db:Gerhard_Scherhorn	datetime:1966
db:Achim_Preiss	datetime:1966
db:Heinz-Willi_Wittschier	datetime:1967
db:Hans_Benninghaus	datetime:1967
db:Herrad_Schenk	datetime:1967/1972
db:Herbert_Homig	datetime:1968
db:Udo_Mainzer	datetime:1968
db:Dietrich_Dickertmann	datetime:1968
db:Klaus_Mullen	datetime:1969
db:Michael_Sachs_(Rechtswissenschaftler)	datetime:1969
db:Ulrich_Menzel	datetime:1969/1974
db:Christian_von_Bar	datetime:1970/1974
db:Gerd_Leuchs	datetime:1970/1975
db:Dieter_H._Stundel	datetime:1971
db:Klaus_Mullen	datetime:1971
db:Klaus_Neidhardt	datetime:1971
db:Hans-Dieter_Heumann	datetime:1972
db:Klaus_Kreiser	datetime:1972
db:Manfred_Melzer	datetime:1972-02-01
db:Heinz_Sahner	datetime:1973
db:Thomas_M._Scheerer	datetime:1973
db:Winfried_Schlepphorst	datetime:1974
db:Thomas_M._Scheerer	datetime:1974
db:Hans_Rupprecht_Goette	datetime:1975/1982
db:Christoph_Antweiler	datetime:1975/1983
db:Rudiger_H._Jung	datetime:1976
db:Norbert_Hanel	datetime:1977/1987
db:Harm_Klueting	datetime:1978
db:Karla_Etschenberg	datetime:1979
db:Martin_Jehne	datetime:1980
db:Frank_Rumscheid	datetime:1980/1990
db:Monika_Schnitzer	datetime:1981
db:Nicholas_Conard	datetime:1982
db:Manfred_Brocker	datetime:1982}-84
db:Karl_Heinz_Lenz	datetime:1983
db:Bruno_Bleckmann	datetime:1983/1989

db:Jens_Claus_Bruning	datetime:1985/1992
db:Christian_Hartmann_(Historiker)	datetime:1986
db:Harald_Zaun	datetime:1986/1993
db:Frank_Glaw	datetime:1987
db:Peter_Geimer	datetime:1987/1992
db:Christoph_Kugelmeier	datetime:1987/1992
db:Martin_Haase	datetime:1988
db:Thomas_Gartner	datetime:1988
db:Christian_Kassung	datetime:1988/1995
db:Karl_Heinz_Lenz	datetime:1990
db:Jurgen_Mittag	datetime:1992
db:Anne_Rothel	datetime:1993
db:Richard_David_Precht	datetime:1994
db:Jurgen_Nielsen-Sikora	datetime:1995/1999
db:Frank_Heidermanns	datetime:1996
db:Thomas_Gartner	datetime:1997/1998
db:Thomas_Gartner	datetime:1998
db:Alex_Meyer	datetime:null
db:Lars_Gohmann	datetime:null
db:Albertus_Magnus	datetime:null
db:Sabrina_van_der_Ley	datetime:null
db:Yann-Benjamin_Kugel	datetime:null
db:Werner_Heinen	datetime:null
db:Hans_Hecker	datetime:null
db:Hans-Joachim_Schumann	datetime:null
db:Johann_Anton_Joseph_Hansen	datetime:null
db:Meister_Eckhart	datetime:null
db:Horst_Albach	datetime:null
db:Andreas_Wilms	datetime:null
db:Theo_Meyer	datetime:null
db:Arsene_Verny	datetime:null
db:Glarean	datetime:null
db:Johann_Machabeus	datetime:null
db:Hans-Christoph_Hobohm	datetime:null
db:Peter_Ulner	datetime:null
db:Tilemann_Stella	datetime:null
db:Dieter_Berg	datetime:null
db:Jakob_Omphal	datetime:null
db:Johann_Jakob_Hemmer	datetime:null
db:Karl_Maria_Hettlage	datetime:null
db:Johannes_Fabri	datetime:null
db:Menso_Alting	datetime:null
db:Gerhard_Westerburg	datetime:null
db:Jean_Ignace_Roderique	datetime:null
db:Adolf_Raskin	datetime:null
db:Ignatz_Stroof	datetime:null

db:Erich_Potthoff	datetime:null
db:Gustav_Adolf_Krieg	datetime:null
db:Ulrich_von_Alemann	datetime:null
db:Reinhold_Viehoff	datetime:null
db:Heinrich_Bullinger	datetime:null
db:Siegfried_Lehrl	datetime:null
db:Gerhart_Baumann	datetime:null
db:Wolfram_Engels	datetime:null
db:Wilhelm_Hoenerbach	datetime:null
db:Horst_Wildemann	datetime:null
db:Max_Morsches	datetime:null
db:Jacob_Montanus	datetime:null
db:Marina_Linares	datetime:null
db:Eberhard_Billick	datetime:null
db:Hermann_Rathmann	datetime:null
db:Dietmar_Kamper	datetime:null
db:Elisabeth_Alfoldi-Rosenbaum	datetime:null
db:Peter_Pooth	datetime:null
db:Elisabeth_Fehrenbach	datetime:null
db:Marion_Glaser	datetime:null
db:Burkhart_Cardauns	datetime:null
db:Bernard_Willms	datetime:null
db:Dieter_Hagermann	datetime:null
db:Hans-Rimbert_Hemmer	datetime:null
db:Johannes_Cladders	datetime:null
db:Wolfgang_Darsow	datetime:null
db:Helmut_van_Thiel	datetime:null
db:Ulrich_Lohmar	datetime:null
db:Hans_Kloft	datetime:null
db:Otto_Kirchheimer	datetime:null
db:Johannes_Sleidanus	datetime:null
db:Helge_Breloer	datetime:null
db:Justus_Sinold	datetime:null
db:Joachim_Starbatty	datetime:null
db:Martin_Jehne	datetime:null
db:Reinhard_Breymayer	datetime:null
db:Reinhard_Feinendegen	datetime:null
db:Franz_Rudolf_Bornewasser	datetime:null
db:Hans_Hohberg	datetime:null
db:Klaus_Pietschmann	datetime:null
db:Hermann_Adam	datetime:null
db:Wilhelm_Hornbostel	datetime:null
db:Armin_Willingmann	datetime:null
db:Klaus_Herbers	datetime:null
db:Ram_Adhar_Mall	datetime:null
db:Albrecht_Encke	datetime:null

db:Wolfgang_Schmid_(Philologe)	datetime:null
db:Caspar_Vopelius	datetime:null
db:Hermann_Pohlmeier	datetime:null
db:Sebastian_Scheerer	datetime:null
db:Peter_von_Mollendorff	datetime:null
db:Hermann_Conrad_(Rechtshistoriker)	datetime:null
db:Karl_Albert_(Philosoph)	datetime:null
db:Georg_Wieland	datetime:null
db:Klaus_Sallmann	datetime:null
db:Jorn_Peter_Hiekel	datetime:null
db:Peter_F._E._Sloane	datetime:null
db:Mark_Benecke	datetime:null
db:Gerhard_Baaken	datetime:null

Ehrendoktoren der Humboldt-Universität zu Berlin, sortiert nach Datum der Verleihung der Ehrendoktorwürde

?s	?g
db:Christian_Ludwig_Ideler	datetime:1814-12-18
db:August_Tholuck	datetime:1822
db:Friedrich_Gustav_Lisco	datetime:1839-11-01
db:Gustav_Rose	datetime:1860-10-16
db:Friedrich_Adler_(Baurat)	datetime:1902
db:Friedrich_Adler_(Baurat)	datetime:1903
db:Hermann_Diels	datetime:1910-10-12
db:Carl_Engler	datetime:1911
db:Johannes_Heckel	datetime:1931
db:Max_Gerlach	datetime:1932
db:Max_Wellmann	datetime:1933-03-15
db:Hugo_Neubauer	datetime:1935
db:Karl_Foerster	datetime:1950
db:Walter_Rothkegel	datetime:1952
db:Josef_Becker-Dillingen	datetime:1954
db:Herbert_Ihering	datetime:1963
db:Werner_Forssmann	datetime:1977
db:George_M._A._Hanfmann	datetime:1981
db:Kurt_Gossweiler	datetime:1988
db:Hans_Meyer_(Jurist)	datetime:1993
db:Friedrich_Hensel_(Physikochemiker)	datetime:1999
db:Friedrich_Hensel_(Physikochemiker)	datetime:2002
db:Carl_Friedrich_Gethmann	datetime:2003
db:Julius_Wess	datetime:2005
db:Elinor_Ostrom	datetime:2007
db:Karl_Ludwig_Gronau	datetime:null
db:Eduard_Maurer	datetime:null
db:Gerhart_Rodenwaldt	datetime:null
db:John_Desmond_Bernal	datetime:null
db:Gunther_Klaffenbach	datetime:null

Deutsche, die in den USA Professoren sind

?s	?g
db:Johann_Christoph_Kunze	datetime:1780
db:August_Konig	datetime:1874
db:Rudolf_Leonhard_(Jurist)	datetime:1907/1908
db:Erich_Marcks_(Historiker)	datetime:1912
db:Jakob_Rosenberg_(Kunsthistoriker)	datetime:1925/1935
db:Carl_Joachim_Friedrich	datetime:1926
db:Carl_Joachim_Friedrich	datetime:1931
db:Hajo_Holborn	datetime:1934
db:Alfred_Rehder	datetime:1934
db:Hans_Adolph_Rademacher	datetime:1934
db:Kurt_Weitzmann	datetime:1935
db:Friedrich_Kessler	datetime:1935/1938
db:Hertha_Sponer	datetime:1936/1966
db:Max_Jakob_(Physiker)	datetime:1937
db:Alexander_Dorner	datetime:1937/1941
db:Kurt_Riezler	datetime:1938
db:Konrad_Bloch	datetime:1938
db:William_Stern	datetime:1938
db:Hugo_Leichtentritt	datetime:1940
db:Arthur_Rosenthal	datetime:1940
db:Otto_Franz_Georg_Schilling	datetime:1943
db:Hugo_Leichtentritt	datetime:1944
db:Maria_Goeppert-Mayer	datetime:1946
db:Konrad_Bloch	datetime:1946
db:Hermann_Gundersheimer	datetime:1947
db:Ludwig_Freund_(Politikwissenschaftler)	datetime:1947
db:Konstantin_Reichardt	datetime:1947
db:Walter_Elsasser	datetime:1947
db:Georg_Joos	datetime:1947-06/Oktober
db:Gregor_Wentzel	datetime:1948
db:Franz_Neumann_(Politikwissenschaftler)	datetime:1948
db:Max_Knoll	datetime:1948/1956
db:Margarete_Bieber	datetime:1949
db:Erwin_Bodky	datetime:1950
db:Hans_Jonas	datetime:1950/1954
db:Hermann_Kellenbenz	datetime:1952/1953
db:Rudolf_Carnap	datetime:1952/1954
db:Hannah_Arendt	datetime:1953
db:Herbert_Marcuse	datetime:1954
db:Wolfgang_Friedmann	datetime:1955
db:Friedrich_Hirzebruch	datetime:1955/1956
db:Franz_Rosenthal	datetime:1956
db:Otto_Franz_Georg_Schilling	datetime:1958

db:Heinz_Eulau	datetime:1958
db:Dieter_Gaier	datetime:1959
db:Hannah_Arendt	datetime:1959
db:Albrecht_Dold	datetime:1960
db:Egon_Sohmen	datetime:1960
db:Rudolf_Haag	datetime:1960/1966
db:Reinhard_Selten	datetime:1961
db:Richard_M._Buxbaum	datetime:1961
db:Arthur_Schweitzer	datetime:1961
db:Christian_Pommerenke	datetime:1961
db:Egon_Sohmen	datetime:1961/1969
db:Arnold_G._Reichenberger	datetime:1961/1973
db:Walter_Wiora	datetime:1962
db:Herbert_Giersch	datetime:1962
db:Joachim_Bumke	datetime:1962
db:Kurt_Schutte	datetime:1962
db:Trutz_Rendtorff	datetime:1962/1968
db:Fritz_Stern	datetime:1963
db:Guy_Stern	datetime:1963
db:Carsten_Colpe	datetime:1963
db:Karl_Friedrich_Stroheker	datetime:1963
db:Hannah_Arendt	datetime:1963/1967
db:Richard_M._Buxbaum	datetime:1964
db:Gisbert_Hasenjaeger	datetime:1964
db:Herbert_Marcuse	datetime:1964
db:John_Rewald	datetime:1964/1971
db:Joachim_Bumke	datetime:1965
db:Ernst-Joachim_Mestmacker	datetime:1965
db:William_Prager	datetime:1965
db:Jurgen_von_Beckerath	datetime:1966
db:Michael_Nagler	datetime:1966/1991
db:Erwin_Rosenthal	datetime:1967
db:Fritz_Stern	datetime:1967
db:Johannes_Renger	datetime:1968
db:Benno_Muller-Hill	datetime:1968
db:Hans_J._Nissen	datetime:1968/1971
db:Dieter_Henrich_(Philosoph)	datetime:1968/1986
db:Trutz_Rendtorff	datetime:1968/1999
db:Arnulf_Baring	datetime:1969
db:Gunter_Harder	datetime:1969
db:Hans_Follmer	datetime:1969/1970
db:Reinhard_Selten	datetime:1969/1972
db:Immo_Appenzeller	datetime:1970
db:Josef_Adolf_Schmoll_genannt_Eisenwerth	datetime:1970
db:Raimund_Apfelbach	datetime:1970/1971
db:Hans_Follmer	datetime:1970/1972

db:Manfred_Wundram	datetime:1970/1989
db:Ingo_Lieb	datetime:1971
db:Wolf-Dieter_Narr	datetime:1971/2002
db:Ulrich_Schindewolf	datetime:1972
db:Gerhard_Gottschalk	datetime:1972/1973
db:Reinhard_Selten	datetime:1972/1984
db:Ralf_Steudel	datetime:1973
db:Bert_Holldobler	datetime:1973/1989
db:Renate_Valtin	datetime:1974
db:Immo_Appenzeller	datetime:1974
db:Hanna_Holborn_Gray	datetime:1974
db:Melitta_Schachner	datetime:1974
db:Josef_Adolf_Schmoll_genannt_Eisenwerth	datetime:1974/1975
db:Friedrich_Hermann_Busse	datetime:1975
db:Henry_Friedlander	datetime:1975/2001
db:Otfried_Hoffe	datetime:1976
db:Melitta_Schachner	datetime:1976
db:Walter_Heiligenberg	datetime:1976
db:Hans-Martin_Barth	datetime:1976/1978
db:Florentine_Mutherich	datetime:1976/1982
db:Jurgen_Gmehling	datetime:1977/1978
db:Jurij_Striedter	datetime:1977/1993
db:Gerhard_Gottschalk	datetime:1978/1979
db:Klaus_Hopt	datetime:1979
db:Uta-Renate_Blumenthal	datetime:1979/1988
db:Roland_Vaubel	datetime:1979}-81
db:Wolfgang_Edelstein	datetime:1980
db:Gert_Bruggemeier	datetime:1980
db:Arnold_Picot	datetime:1980
db:Uwe_H._Schneider	datetime:1981
db:Henning_Kohler_(Historiker)	datetime:1981
db:Joachim_Cuntz	datetime:1982/1985
db:Frederic_Vester	datetime:1982/1989
db:Walter_Kasper	datetime:1983
db:Wolfgang_Brezinka	datetime:1983
db:Thomas_Pogge	datetime:1983/2006
db:Gerd_Gigerenzer	datetime:1984/1990
db:Wolfgang_Huber	datetime:1984/1994
db:Ulrich_Gosele	datetime:1985
db:Peter_Richter_(Physiker)	datetime:1985/1986
db:Helmut_Tributsch	datetime:1985/1986
db:Volker_Weispfenning	datetime:1986
db:Hans_Joas	datetime:1986
db:Detlev_Poguntke	datetime:1986
db:Friedrich-Karl_Thielemann	datetime:1986
db:Josef_Adolf_Schmoll_genannt_Eisenwerth	datetime:1986

db:Wolfgang_Hardtwig	datetime:1987
db:Heinrich_Oberreuter	datetime:1987
db:Adolf_Seilacher	datetime:1987
db:Wolfgang_Daubler	datetime:1987
db:Wolfgang_Sachs	datetime:1987/1990
db:Thomas_Wolff_(Chemiker)	datetime:1988
db:Andreas_Floer	datetime:1988
db:Josef_Adolf_Schmoll_genannt_Eisenwerth	datetime:1988
db:Jurgen_Renn	datetime:1989
db:Georg_Bollenbeck	datetime:1989
db:Josef_Adolf_Schmoll_genannt_Eisenwerth	datetime:1989
db:Benjamin_Buchloh	datetime:1989/1994
db:Axel_Borsch-Supan	datetime:1989/2011
db:Wolfgang_Luck	datetime:1990
db:Wilhelm_Gruissem	datetime:1990
db:Thomas_G._Rosenmeyer	datetime:1990
db:Martin_Lohse	datetime:1990
db:Berndt_Mueller	datetime:1990
db:Hans_Ulrich_Buhl	datetime:1990/1994
db:Gerhard_Huisken	datetime:1991
db:Joachim_Maier_(Chemiker)	datetime:1991
db:Paul_Knochel	datetime:1991
db:Ellen_M._Immergut	datetime:1991/1994
db:Gunter_Blobel	datetime:1992
db:Baber_Johansen	datetime:1992/1994
db:Ulrike_Gaul	datetime:1993
db:Armin_Wolf_(Historiker)	datetime:1993
db:Friedrich_Hirzebruch	datetime:1993
db:Michael_Brenner_(Historiker)	datetime:1993/1994
db:Christian_Haass	datetime:1993/1995
db:Manfred_Strecker	datetime:1993/1995
db:Karl-Joachim_Holkeskamp	datetime:1994
db:Wolfgang_Daubler	datetime:1994
db:Bernd_Sturmfels	datetime:1995
db:Wolfgang_Daubler	datetime:1995
db:Friedrich_Kittler	datetime:1996
db:Gert_Bruggemeier	datetime:1996/1997
db:Arnold_Angenendt	datetime:1997
db:Michael_Bolle	datetime:1997/1998
db:Peter_Duesberg	datetime:1997/2000
db:Gustav_Gerber	datetime:1998
db:Volker_Berghahn	datetime:1998
db:Arnold_Angenendt	datetime:1999
db:Kaspar_Elm	datetime:1999
db:Gert_Bruggemeier	datetime:1999
db:Hans_Joas	datetime:2000

db:Georg_Bollenbeck	datetime:2000
db:Joachim_Stohr	datetime:2000
db:Ursula_Staudinger	datetime:2000
db:Thomas_von_Danwitz	datetime:2000/2001
db:Dirk_Wentzel	datetime:2000/2002
db:Hans-Josef_Klauck	datetime:2001
db:Stefan_R._Hauser	datetime:2001
db:Bernhard_Giesen	datetime:2001
db:Andreas_Rodder	datetime:2001-10/September
db:Stefanie_Petermichl	datetime:2002
db:Nina_Zimmer	datetime:2002
db:Georg_Bollenbeck	datetime:2002
db:Michael_Minkenberg	datetime:2003
db:Olaf_Muller_(Philosoph)	datetime:2003
db:Gunter_Harder	datetime:2003
db:Markus_Asper	datetime:2003
db:Reinhard_Zollner	datetime:2003/2004
db:Barbara_Vinken	datetime:2004
db:Nicola_Fuchs-Schundeln	datetime:2004
db:Matthias_Schundeln	datetime:2004
db:Georg_Bollenbeck	datetime:2004
db:Burkhard_Meissner	datetime:2004
db:Farouk_Grewing	datetime:2004
db:Tom_Abel	datetime:2004
db:Markus_Asper	datetime:2004
db:Gert_Bruggemeier	datetime:2004-10/Februar
db:Jorg_Baten	datetime:2005
db:Baber_Johansen	datetime:2005
db:Benjamin_Buchloh	datetime:2005
db:Markus_Greiner	datetime:2005-08
db:Jost_Dulffer	datetime:2005/2006
db:Henning_Schmidgen	datetime:2005/2006
db:Joseph_Vogl	datetime:2005/2006
db:Jorg_Baten	datetime:2006/2007
db:Joseph_Vogl	datetime:2007
db:Wolfgang_Schleich	datetime:2008
db:Wolfram_Knauer	datetime:2008
db:Karl_Lauterbach_(Politiker,_1963)	datetime:2008
db:Rudiger_Minor	datetime:2008
db:Martin_Bojowald	datetime:2009
db:Helmut_Muller-Enbergs	datetime:2010
db:Horst_Eidenmuller	datetime:2011
db:Wilhelm_Schmidt-Biggemann	datetime:2011
db:Andreas_Rodder	datetime:????-04
db:Ursula_Staudinger	datetime:Mai/2003-07
db:Kurt_Riezler	datetime:null

```
db:Nikolaus_Rajewsky                              datetime:null
db:Tom_Abel                                       datetime:null
db:Paul_Klein                                     datetime:null
db:Albert_Ziegler_(Psychologe)                    datetime:null
db:Peter_Gritzmann                                datetime:null
db:Hans_Joas                                      datetime:null
db:Karl_Ziegler_(Chemiker)                        datetime:null
db:Alf_Ludtke                                     datetime:null
db:Heiko_Uecker                                   datetime:null
db:Dirk_Kreimer                                   datetime:null
db:Bernd_Sturmfels                                datetime:null
db:Peter_Duesberg                                 datetime:null
db:Richard_Goldschmidt                            datetime:null
db:Ossip_K._Flechtheim                            datetime:null
db:Gregor_Schollgen                               datetime:null
db:Volker_Berghahn                                datetime:null
db:Carl_Wolfgang_Muller                           datetime:null
db:Horst_Fischer_(Jurist)                         datetime:null
db:Wolfgang_Friedmann                             datetime:null
db:Johann_Christoph_Kunze                         datetime:null
db:Katharina_Volk                                 datetime:null
db:Johann-Matthias_Graf_von_der_Schulenburg       datetime:null
db:Hans_Horst_Meyer                               datetime:null
db:Gunter_Blobel                                  datetime:null
db:Henry_Friedlander                              datetime:null
db:Juan_Linz                                      datetime:null
db:Egon_Sohmen                                    datetime:null
db:Walter_Greiner                                 datetime:null
db:Hans_von_Hentig                                datetime:null
db:Spiros_Simitis                                 datetime:null
db:Bernhard_Korte                                 datetime:null
db:Hans_Jacob_Reissner                            datetime:null
db:Erik_H._Erikson                                datetime:null
db:Marie_Munk                                     datetime:null
db:Herbert_Bloch                                  datetime:null
db:Winfried_Nerdinger                             datetime:null
db:Karl_Vietor                                    datetime:null
db:Georg_Nicolaus_Knauer                          datetime:null
db:August_Konig                                   datetime:null
db:Hans_Neisser                                   datetime:null
db:Shlomo_Dov_Goitein                             datetime:null
db:Arthur_R._von_Hippel                           datetime:null
db:Winfried_Fluck                                 datetime:null
db:Helmut_H._Schaefer                             datetime:null
db:Hans_Bock_(Chemiker)                           datetime:null
db:Jurgen_Trabant                                 datetime:null
```

db:Thorsten_Hens	datetime:null
db:Hans_Ulrich_Gumbrecht	datetime:null
db:Joachim_Schwalbach	datetime:null
db:Hermann_Frankel	datetime:null
db:Manfred_Strecker	datetime:null
db:Irmgard_Lotz	datetime:null
db:Peter_Losche	datetime:null
db:Hans-Gunter_Rolff	datetime:null
db:Sebastian_Thrun	datetime:null
db:Rainer_Hertel	datetime:null
db:Albrecht_Cordes	datetime:null
db:Karl_Korsch	datetime:null
db:Hans_J._Muller-Eberhard	datetime:null
db:Karl_Lehmann_(Archaologe)	datetime:null
db:Oscar_Weigert	datetime:null
db:Wichard_Woyke	datetime:null
db:Heribert_Meffert	datetime:null
db:Hajo_Funke	datetime:null

Personen, die zwischen 1920 und 1929 in Hannover aktiv waren

?s	?p	?o	?stadt	?g
db:Georg_Schnath	prop:has_studied_at	db:Kaiser-Wilhelm-_und_Ratsgymnasium_Hannover	db:Hannover	datetime:1917/1922
db:Conrad_Muller	prop:lives_in	NULL	db:Hannover	datetime:1919/1923
db:Alexander_Dorner	prop:is_lecturer_at	db:Gottfried_Wilhelm_Leibniz_Universitat_Hannover	db:Hannover	datetime:1920/1937
db:Ludwig_Kiepert	prop:lives_in	NULL	db:Hannover	datetime:1921
db:Heinrich_Dorrie	prop:has_studied_at	db:Kaiser-Wilhelm-_und_Ratsgymnasium_Hannover	db:Hannover	datetime:1921
db:Christoph_Steding	prop:has_studied_in	NULL	db:Hannover	datetime:1922
db:Bruno_Schulz_(Architekt)	prop:is_honorary_doctor_at	db:Gottfried_Wilhelm_Leibniz_Universitat_Hannover	db:Hannover	datetime:1922
db:Horst_von_Sanden	prop:is_professor_at	db:Gottfried_Wilhelm_Leibniz_Universitat_Hannover	db:Hannover	datetime:1922
db:Axel_Schur	prop:is_lecturer_at	db:Gottfried_Wilhelm_Leibniz_Universitat_Hannover	db:Hannover	datetime:1923
db:Herbert_Backe	prop:works_at	db:Gottfried_Wilhelm_Leibniz_Universitat_Hannover	db:Hannover	datetime:1923/1924
db:Rudolf_Zurmuhl	prop:has_studied_in	NULL	db:Hannover	datetime:1924
db:Peter_Danckwortt	prop:is_professor_at	db:Tierarztliche_Hochschule_Hannover	db:Hannover	datetime:1924
db:Erich_Koestermann	prop:has_studied_at	db:Gottfried_Wilhelm_Leibniz_Universitat_Hannover	db:Hannover	datetime:1924
db:Konrad_Ludwig_(Mathematiker)	prop:works_at	db:Gottfried_Wilhelm_Leibniz_Universitat_Hannover	db:Hannover	datetime:1924-05/1935
db:Gunther_Schiemann	prop:works_at	db:Gottfried_Wilhelm_Leibniz_Universitat_Hannover	db:Hannover	datetime:1926
db:Gunther_Schiemann	prop:is_lecturer_at	db:Gottfried_Wilhelm_Leibniz_Universitat_Hannover	db:Hannover	datetime:1926
db:Harald_Schering	prop:works_in	NULL	db:Hannover	datetime:1927-04-01
db:Walter_Grossmann	prop:works_at	db:Gottfried_Wilhelm_Leibniz_Universitat_Hannover	db:Hannover	datetime:1928
db:Friedrich_Quincke	prop:is_honorary_doctor_at	db:Tierarztliche_Hochschule_Hannover	db:Hannover	datetime:1928
db:Alexander_Dorner	prop:is_professor_at	db:Gottfried_Wilhelm_Leibniz_Universitat_Hannover	db:Hannover	datetime:1928
db:Wilhelm_Pessler	prop:works_in	NULL	db:Hannover	datetime:1928/1945
db:Erich_Klinge_(Sportwissenschaftler)	prop:is_professor_in	NULL	db:Hannover	datetime:1929
db:Georg_Hoeltje	prop:has_place_of_phd	db:Gottfried_Wilhelm_Leibniz_Universitat_Hannover	db:Hannover	datetime:1929
db:Georg_Hoeltje	prop:works_at	db:Gottfried_Wilhelm_Leibniz_Universitat_Hannover	db:Hannover	datetime:1929-04-01

Personen, die an derselben Hochschule studiert haben, an der sie später Professoren wurden

1469 Treffer, hier aufgrund der hohen Zahl nicht dargestellt.

Die Ergebnisse sind in der Datei `5-gleiche-hochschule-studium-professur.result` im Verzeichnis `queries/` auf der beiliegenden CD-ROM enthalten.

Personen, die an einer Hochschule mit bis zu 5000 Studenten studiert haben und Professoren wurden an einer Hochschule ab 30000 Studenten

?s	?stud	?prof
db:Gustav_Hegi	db:Universitat_Zurich	db:Ludwig-Maximilians-Universitat_Munchen
db:Andreas_Kappeler	db:Universitat_Zurich	db:Universitat_Wien
db:Walter_Hollstein	db:Universitat_Basel	db:Westfalische_Wilhelms-Universitat
db:Victor_Klemperer	db:Universitat_Genf	db:Technische_Universitat_Dresden
db:Karl_Hermann_Spitzy	db:Gustav-Siewerth-Akademie	db:Universitat_Wien
db:Jean_Taylor	db:University_of_Warwick	db:Rutgers_University
db:Michael_Metzeltin	db:Universitat_Basel	db:Universitat_Wien
db:Hans_Kniep	db:Universitat_Genf	db:Universitat_Strassburg
db:Ulrich_Battis	db:Deutsche_Hochschule_fur_Verwaltungswissenschaften_Speyer	db:Universitat_Hamburg
db:Guido_Seiler	db:Universitat_Zurich	db:University_of_Manchester
db:Hermann_Mooser	db:Universitat_Lausanne	db:Nationale_Autonome_Universitat_von_Mexiko
db:Arthur_Schweitzer	db:Universitat_Basel	db:Freie_Universitat_Berlin
db:Wolfgang_Klaui	db:Universitat_Zurich	db:RWTH_Aachen
db:Franz_Xaver_Bischof	db:Universitat_Luzern	db:Ludwig-Maximilians-Universitat_Munchen
db:Michael_Landmann	db:Universitat_Basel	db:Freie_Universitat_Berlin
db:Martin_Paul_Wassmer	db:Universitat_Lausanne	db:Universitat_zu_Koln
db:Chuu-Lian_Terng	db:Brandeis_University	db:University_of_California,_Berkeley
db:Cleve_Moler	db:California_Institute_of_Technology	db:University_of_Michigan
db:Gordon_Willis_Williams	db:Trinity_College_(Dublin)	db:University_of_California,_Berkeley
db:Fritz_Sturm_(Jurist)	db:Universitat_Lausanne	db:Johannes_Gutenberg-Universitat_Mainz
db:Fritz_Sturm_(Jurist)	db:Universitat_Genf	db:Johannes_Gutenberg-Universitat_Mainz
db:Sergio_Albeverio	db:ETH_Zurich	db:Ruhr-Universitat_Bochum
db:Robert_Schweizer	db:Universitat_Lausanne	db:Ludwig-Maximilians-Universitat_Munchen
db:Thomas_Schreijack	db:Universitat_Basel	db:Johann-Wolfgang-Goethe-Universitat_Frankfurt_am_Main
db:Markus_Kotzur	db:Duke_University	db:Universitat_Hamburg
db:Therese_Fuhrer	db:Universitat_Basel	db:Freie_Universitat_Berlin
db:Emile_Picard	db:Ecole_polytechnique	db:Universitat_Toulouse
db:Johann_Otto_Tabor	db:Universitat_Genf	db:Universitat_Strassburg
db:Max_Born	db:Universitat_Zurich	db:Universitat_Breslau
db:Ernst-Ludwig_Winnacker	db:ETH_Zurich	db:Ludwig-Maximilians-Universitat_Munchen
db:Hans_Steinhart	db:ETH_Zurich	db:Universitat_Hamburg

db:Robert_Moody	db:University_of_Saskatchewan	db:University_of_Alberta
db:Robert_Tarjan	db:California_Institute_of_Technology	db:New_York_University
db:Roland_Wiesendanger	db:Universitat_Basel	db:Universitat_Hamburg
db:David_Bohm	db:California_Institute_of_Technology	db:Universitat_von_Sao_Paulo
db:Armin_Steinkamm	db:Universitat_Genf	db:Ludwig-Maximilians-Universitat_Munchen
db:Thomas_G._Rosenmeyer	db:School_of_Oriental_and_African_Studies	db:University_of_California,_Berkeley
db:Thomas_Immoos	db:Universitat_Zurich	db:Universitat_Wien
db:Bettina_Heintz	db:Universitat_Zurich	db:Universitat_Wien
db:Bettina_Heintz	db:Universitat_Zurich	db:Johannes_Gutenberg-Universitat_Mainz
db:Hans_Heinrich_Landolt	db:Universitat_Zurich	db:Universitat_Strassburg
db:Dietrich_Schindler_junior	db:Universitat_Zurich	db:University_of_Michigan
db:Manfred_A._Dauses	db:Universitat_Lausanne	db:Karls-Universitat_Prag
db:Hanspeter_Kraft	db:Universitat_Basel	db:Universitat_Hamburg

Deutsch-französische Zusammenarbeit

?a	?b
db:Norbert_Schappacher	db:Catherine_Goldstein
db:Catherine_Goldstein	db:Norbert_Schappacher
db:Jean_Dieudonne	db:Alexander_Grothendieck
db:Charles-Victor_Mauguin	db:Carl_Hermann_(Physiker)
db:Carl_Einstein	db:Jean_Renoir
db:Richard_Assmann	db:Leon-Philippe_Teisserenc_de_Bort
db:Robert_Bourgeois	db:Philipp_Furtwangler_(Mathematiker)
db:Joseph_Louis_Gay-Lussac	db:Justus_von_Liebig
db:John_Scheid	db:Jorg_Rupke
db:Walter_Borho	db:Jean-Luc_Brylinski
db:Walter_Borho	db:Pierre_Gabriel
db:Horst_Leuchtmann	db:Pierre_Bertaux
db:Gunther_Roeder	db:Gaston_Maspero
db:Otto_Honigschmid	db:Henri_Moissan
db:Max_Wolff	db:Ernst_Stahl_(Botaniker)
db:Georg_Anschutz	db:Alfred_Binet
db:Barthel_Hrouda	db:Jean_Bottero
db:Pierre_Judet_de_la_Combe	db:Heinz_Wismann_(Altphilologe)
db:Pierre_Judet_de_la_Combe	db:Gregor_Vogt-Spira
db:Henri_Cohen	db:Gerhard_Frey_(Mathematiker)
db:Wilhelm_Vleugels	db:Gustave_Le_Bon
db:Pierre_Gabriel	db:Alexander_Grothendieck
db:Theo_Sundermeier	db:Emmanuel_Levinas
db:Joachim_Schwermer	db:Catherine_Goldstein
db:Ernst_Stahl_(Botaniker)	db:Julius_Sachs
db:Carl_Lowig	db:Antoine-Jerome_Balard
db:Marc_Boegner	db:Frederik_J._Forell
db:Henry_Darcy	db:Julius_Weisbach
db:Gaston_Maspero	db:Emil_Brugsch
db:Carl_Hermann_(Physiker)	db:Charles-Victor_Mauguin
db:Friedrich_Kittler	db:Jean_Baudrillard
db:Friedrich_Kittler	db:Jacques_Derrida
db:Tania_Singer	db:Matthieu_Ricard
db:Henri_Etienne_Sainte-Claire_Deville	db:Friedrich_Wohler
db:Emil_Brugsch	db:Gaston_Maspero
db:Ernst_Nolte	db:Francois_Furet
db:Reinhold_Koser	db:Pierre-Louis_Moreau_de_Maupertuis
db:Luc_Illusie	db:Alexander_Grothendieck
db:Manfred_Clauss	db:Franck_Goddio
db:Alfred_Kastler	db:Jean_Brossel
db:Horst_Heilmann	db:Napoleon_Bonaparte
db:Manes_Sperber	db:Willi_Munzenberg
db:Jacques_Herbrand	db:Emmy_Noether
db:Jacques_Herbrand	db:Helmut_Hasse